IDA PAPER P-3003

GOVERNMENT-SPONSORED RESEARCH AND DEVELOPMENT EFFORTS IN THE AREA OF INTELLIGENT TUTORING SYSTEMS

Christine Youngblut

September 1994

Approved for public release, unlimited distribution: January 11, 1995.

INSTITUTE FOR DEFENSE ANALYSES
1801 N. Beauregard Street, Alexandria, Virginia 22311-1772

DEFINITIONS

IDA publishes the following documents to report the results of its work.

Reports

Reports are the most authoritative and most carefully considered products IDA publishes. They normally embody results of major projects which (a) have a direct bearing on decisions affecting major programs, (b) address issues of significant concern to the Executive Branch, the Congress and/or the public, or (c) address issues that have significant economic implications. IDA Reports are reviewed by outside panels of experts to ensure their high quality and relevance to the problems studied, and they are released by the President of IDA.

Group Reports

Group Reports record the findings and results of IDA established working groups and panels composed of senior individuals addressing major issues which otherwise would be the subject of an IDA Report. IDA Group Reports are reviewed by the senior individuals responsible for the project and others as selected by IDA to ensure their high quality and relevance to the problems studied, and are released by the President of IDA.

Papers

Papers, also authoritative and carefully considered products of IDA, address studies that are narrower in scope than those covered in Reports. IDA Papers are reviewed to ensure that they meet the high standards expected of refereed papers in professional journals or formal Agency reports.

Documents

IDA Documents are used for the convenience of the sponsors or the analysts (a) to record substantive work done in quick reaction studies, (b) to record the proceedings of conferences and meetings, (c) to make available preliminary and tentative results of analyses, (d) to record data developed in the course of an investigation, or (e) to forward information that is essentially unanalyzed and unevaluated. The review of IDA Documents is suited to their content and intended use.

The work reported in this document was conducted under IDA's Central Research Program. Its publication does not imply endorsement by the Department of Defense or any other Government Agency, nor should the contents be construed as reflecting the official position of any Government Agency.

UNCLASSIFIED

IDA PAPER P-3003

GOVERNMENT-SPONSORED RESEARCH AND DEVELOPMENT EFFORTS IN THE AREA OF INTELLIGENT TUTORING SYSTEMS

Christine Youngblut

September 1994

INSTITUTE FOR DEFENSE ANALYSES

IDA Central Research Program 9000-542

UNCLASSIFIED

PREFACE

This document presents the data collected in the course of a Central Research Project, Information Technology for Education and Training, that reports on current Government-sponsored research and development activities in the intelligent tutoring systems field. The analysis of the data will be reported separately in a later document. To make the data available in a timely manner to researchers in this field, it has been decided not to delay the publication of this document pending the completion of the analysis report.

The author gratefully acknowledges the support provided by all those who took time from their busy schedules to discuss their work. In particular, special thanks go to the following people:

Dr. John Anderson
Mrs. Carol Barclay
Dr. Susan Chipman
Ms. Susan Dahl
Dr. Sharon Derry
Dr. Jack Dohme
Dr. Martha Evens
Mr. Jim Fleming
Dr. John Gluckman
Dr. Ellen Hall
Dr. Fred Hofstetter
Dr. Willard Holmes
Dr. Laveen Kanal
Dr. Jeffrey Kantor
Dr. John Kaplan
Dr. Sandy Katz
Mr. Tom Kopke
Dr. John Leddo
Dr. Hugh Bowen Loftin
Dr. Jean MacMillan
Mr. Tim Maher
Dr. John Maguire
Dr. Jim Mazzu
Dr. David McArthur
Dr. Andy Molnar
Dr. Johanna Moore
Dr. Jack Mostow
Ms. Susan Moulton
Mr. John Neal
Lt. Col. Jim Parlett
Dr. Ray Perez
Mr. Admiral Piper
Dr. Bruce Porter
Dr. Joe Psotka
Dr. Wes Regian
Dr. Ely Robinson
Dr. John Schuler
Dr. Bob Seidel
Mr. Sid Sittar
Dr. Bob Smillie
Dr. Robert Snuggs
Ms. Paula Watson
Dr. Kent Williams
Dr. Ruth Willis
Dr. Bev Woolf
Dr. Greg Zacharias

Table of Contents

List of Figures

List of Figures

List of Tables

1. INTRODUCTION

This document provides a snapshot of various Government-sponsored intelligent tutoring system (ITS) research and development (R&D) efforts that are underway at this time. The information was collected in the course of reporting on current activities in the ITS field and is presented here as a potential information resource for researchers in this field.

The following Government organizations were contacted and provided information:

- Department of Defense (DOD).
  - Air Force Armstrong Laboratory, Human Resources Directorate.
  - Air Force Human Systems Center, Human Systems Program Office.
  - Army Materiel Command, Missile Command (MICOM), Research Development and Engineering Center (RDEC), Guidance and Control Directorate (G&CD).
  - Army Simulation, Training and Instrumentation Command (STRICOM)
  - Army Research Institute (ARI), Advanced Training Methods Research Unit.
  - Army Research Institute, Aviation R&D Activity (ARIARDA).
  - Defense Modeling and Simulation Office (DMSO).
  - Office of Naval Research (ONR), Science and Technology Directorate.
  - Navy Personnel Research and Development Center (NPRDC).
  - Naval Air Warfare Center Training Systems Division (NAWCTSD).
  - Space and Naval Warfare Systems Command (SPAWAR), Undersea Surveillance Division.

- Department of Education (DOE), Fund for Improvement of Postsecondary Education (FIPSE) Program.
- National Aeronautics and Space Administration (NASA), Johnson Space Center, Software Technology Branch.
- National Science Foundation (NSF), Advanced Technologies Program.
- Technology Reinvestment Project.

The following sections of this document address each of these organizations in turn. In general, each section starts by outlining the *Mission and Role* of the organization, and a *Summary of Past ITS-Related Work* is followed by a overview of the *Overall ITS R&D Program*. However, this information is not uniformly available. For example, in some cases, an organization is undertaking its first ITS-related effort, has had no prior projects in the area, and, as yet, has no formal R&D plan for work in this area.

In each section, these background discussions are followed by descriptions of ongoing projects and any future projects that are already in advanced planning stages. Where the available information permitted it, a standard format has been used to present the project descriptions. The topics discussed are as follows:

Programmatic Background
Planned Products
Prior Related Work
Approach
Potential Follow-On Work

Some of these projects are engaged in basic research and, at this time, are not developing an actual ITS. Most of the projects, however, do include the development of an intelligent tutor, or in some cases, of several tutors. Description of these ITS are included where available. Here again, an attempt was made to follow a standard format, centering on the following topics:

Development Status
Architecture
Evaluation Status
Operating Environment
Future System Plans

2. ARMSTRONG LABORATORY

2.1 Mission and Role of the Human Resources Directorate

The mission of the Air Force Armstrong Laboratory[1], Human Resources Directorate (AL/HR) is to perform scientific research, to develop technologies and improve methods for integrating people with weapons and systems to optimize the human role in Air Force combat effectiveness, and to provide research results and technology to the scientific community, Government organizations, and industry.

AL/HR has played a major role in the development of ITS for over 15 years. Recently, this role been greatly extended. In accordance with Defense Science & Technology Reliance (see Figure 1 on page 4), the Air Force has responsibility for defense-sponsored R&D in Intelligent Computer-Aided Training. Work in this technology subarea as a whole is now performed under the auspices of AL/HR. ITS work, in particular, is the responsibility of the Technical Training Research Division of AL/HR which has the following objectives:

- Demonstrate intelligent tutoring systems.
- Improve technical skills training and development.
- Improve training evaluation systems.
- Improve training decision support systems.

Accordingly, AL/HR has major programs in the areas of intelligent tutoring systems, computer-based training, training and performance measurement, training decision support technology, and job-aiding/training allocation technology.

2.2 Summary of Past ITS-Related Work

AL/HR has sponsored numerous projects addressing topics ranging from knowledge acquisition for ITS to low-cost/rapid ITS development. Application domains for ITS have ranged from orbital mechanics to post-operative medical care, and researchers have

---

[1] Formerly the Air Force Human Resources Laboratory.

In 1989, the Deputy Secretary of Defense's Defense Management Report initiative challenged the Services to create a new approach to science and technology (S&T) management that would increase efficiency and reduce unwarranted overlap in military research, development, test, and evaluation. During 1898-1990, the Tri-Service S&T Reliance initiative examined opportunities to consolidate and collocate R&D efforts at single-site locations in selected technology areas. Reliance is the most comprehensive restructuring effort involving technology base R&D in over 40 years.

Reliance currently has identified 31 technology areas, plus basic research, that are of importance to two or more Reliance participants, and has brought activities in each of these areas under joint planning. There are four major management oversight bodies: the Joint Director of Laboratories (JDL); the Armed Services Biomedical Research, Evaluation and Management Committee (ASBREM); the Training and Personnel Systems Science and Technology Evaluation and Management Committee (TAPSTEM); and the Joint Engineers. They are responsible for the technology areas shown below.

Joint Engineers
Civil Engineering
Environmental Quality

ASBREM
Medical

TAPSTEM
Manpower & Personnel
Training Systems

JDL
Advanced Materials
Aeropropulsion
Air Vehicles (fixed wing)
Air Vehicles (rotary)
Astrometry
Chemical & Biological Defense
Clothing, Textiles & Food
Command, Control & Communications
Computer Sciences
Conventional Air/Surface Weaponry
Crew Systems
Directed Energy Weaponry
Electro-Optics
Electronic Devices
Electronic Warfare
Environmental Sciences
Explosive Ordnance Disposal
Fuels & Lubricants
Ground Vehicles
Integrated Avionics
Nuclear Weapons Effects
Radar
Ships/Watercraft
Small Arms
Space Vehicles
Unmanned Ground Vehicles
Basic Research

To comply with Reliance in the Manpower and Personnel and Training (MPT) systems areas, the TAPSTEM organization was formally established in November 1990 by a joint service agreement. TAPSTEM has four objectives:

- To address organizational roles and resolve service organizational/functional alignment issues.
- To ensure program relevance and obviate duplication via a timely review process.
- To define service issues that require resolution/coordination with other federal agencies outside TAPSTEM.

The Service locations for the MPT systems technology areas, identifying Reliance technology subareas, are shown below. The Air Force has been given responsibility for Intelligent Computer-Aided Training, located at Brooks AFB, Armstrong Laboratory, Human Resources Directorate. The Navy has placed a civilian in Brooks to perform R&D on Intelligent Computer-Aided Training.

Army
- Selection and Classification
- Service Unique Applications
- Human Resources Development
- Land Warfare/Rotary Wing Training
- Unit Collective Training

Air Force
- Selection and Classification
  - Basic Abilities Testing
  - Job Structures & Requirements
  - Service Unique Applications
  - Air Warfare Training Effectiveness
- Intelligent Computer-Aided Training

Navy
- Selection and Classification
  - Computer-based Entrance Testing
  - Service Unique Applications
- Force Management and Modeling
- Sea Warfare Training
- Training Devices and Instructional Features
- Classroom Training
- Productivity Measurement & Enhancement

Figure 1. Project Reliance, TAPSTEM and Responsibilities for Intelligent Training

investigated issues ranging from evaluating the relative training effectiveness of various approaches for automated training of high performance skills, to team-oriented training. This work has had many notable successes. For example, researchers at AL/HR designed, developed, and evaluated both laboratory and field versions of the Word Problem Solving (WPS) tutor in support of the Air Force's Fundamental Skills Training (FST) research project. Not only was this the largest field study of its kind ever attempted, but Version 3.0 of WPS produced average gains of 26% in problem-solving performance among the treatment population—an outcome that is both highly significant and highly reliable.

Further, AL/HR work has resulted in several examples of Air Force-developed dual-use ITS technology:

- Fuel Cell Emergency Procedures Tutor. Built for DOD, used to train NASA Space Shuttle crews/ground crews.
- Challenger Space Operations Console Tutor. Built through SBIR, used by DOD, and commercialized for sale to universities and industry.
- Fundamental Skills Training (FST) tutors for ninth grade classrooms. Transfers principles of ITS technology to public education customers.
- Microcomputer Intelligence for Technical Training authoring shell (MITT Writer). Produces low-cost ITSs quickly (in less than six months, for less than $50K).

Transition of ITS technology is currently underway to Air Combat Command, Air Force Special Operations Command, Space Command, Air Education and Training Command, and Navy Technical Training at Mare Island Shipyards.

2.3 Overall ITS R&D Program

The AL/HR ITS research program has been designed to reflect two major concerns. First, in order to manage the trade-off between internal and external ITS validity, ITS R&D progresses from laboratory studies of pedagogy in artificial tasks toward field studies of fully implemented ITS for real-world tasks. This allows starting with the identification of powerful instructional manipulations in controlled settings, and then working up to evaluating those manipulations in applied settings. The second concern is to maximize the efficiency of the research as well as the generality of results. This has caused ITS R&D to be driven by learning theory and constrained by evaluation data. This overall strategy is depicted in Figure 2 on page 6.

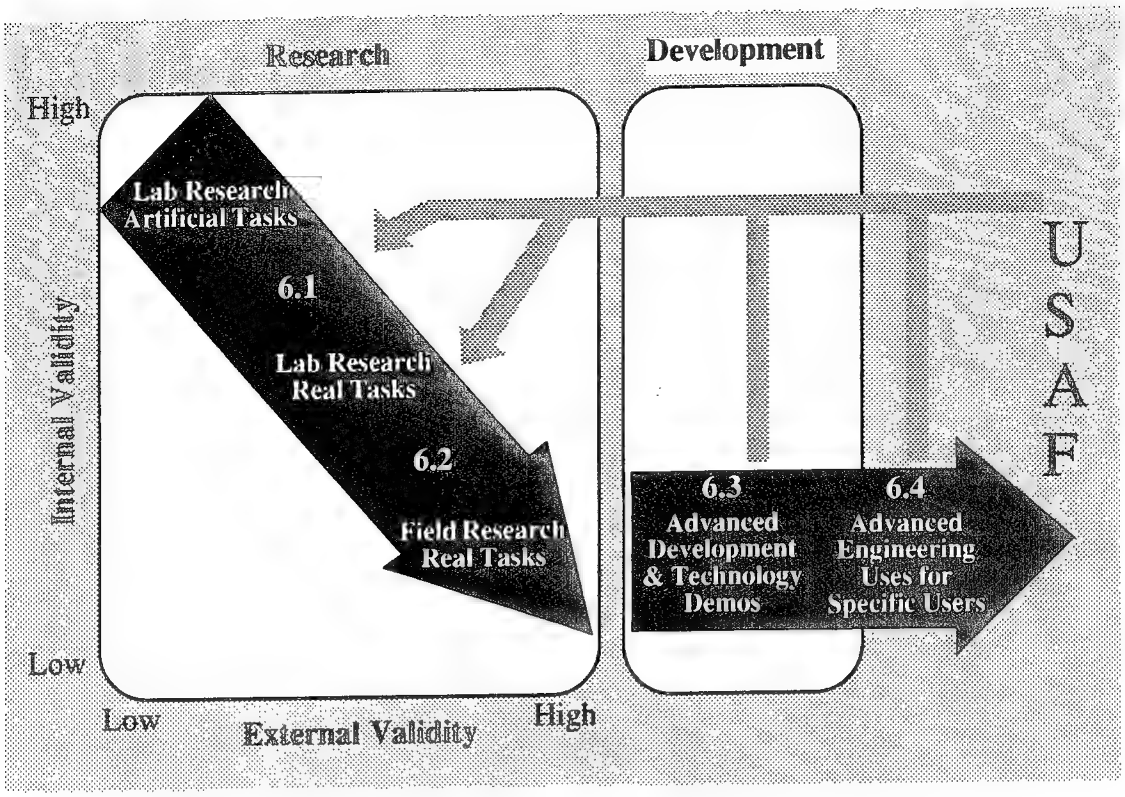


Figure 2. Automated Instruction: Long-Term R&D Strategy

2.4 On-Going ITS-Related Tasks

The AL/HR ITS research and development strategy has three interrelated parts which correspond to three major projects:

- Basic Research (6.1): Training Research for Automated Instruction (TRAIN). Conducts laboratory-based research into transferring principles of human instruction into principles of automated instruction.
- Exploratory Research (6.2): Fundamental Skills Training (FST). Develops prototype dual-use ITSs for test and evaluation in secondary school field settings.
- Advanced Development (6.3): Intelligent Computer-Assisted Training Testbeds (ICATT). Applies rapid prototyping principles to the development of ITS authoring shells which dramatically reduce ITS development time and cost.

These projects are discussed in more detail below, followed by descriptions of other, smaller projects.

2.4.1 Training Research for Automated Instruction (TRAIN) Project

The goal of the TRAIN project is to understand the relationship among automated pedagogies, personological variables, and instructional effectiveness. It is a long-term, large-scale project that will investigate theoretical cognitive approaches in automated education and training, as well as provide empirical documentation of well-specified curriculum approaches for specific desired learning outcomes. Following are examples of research questions being asked:

- What is an appropriate taxonomic characterization of tasks that will optimally support generalizations about research findings on pedagogy and subsequently support task analysis targeted for the design of training interventions?
- What is the appropriate role of microprocessors in training? Which tasks are amenable to automated instruction and what pedagogies are most efficient for those tasks?
- How should automated curricula be designed in order to minimize the negative effects of massed instruction on performance, since most training centers use massed rather than distributed practice sessions?
- How can protocols be developed which allow multiple students to learn simultaneously from a single computer, since many training centers have too few computers?

To support these broad goals, a computer laboratory has been built (called the Cooperative Lab, or Co-Lab), comprising 30 training delivery microcomputers and assets to support 50,000 plus subject hours per year.

Research on pedagogy conducted solely in the laboratory or solely in the field can be subject to serious deficiencies. Laboratory research can have high internal validity (for example, good experimental control and elimination of confounds) but is subject to low external validity (ability to generalize to applied settings). Field research can have high external validity but is subject to low internal validity. The AL/HR solution to this problem has been to develop the capability to conduct both laboratory and field research. For example, at the Co-Lab, researchers are able to conduct tightly controlled studies, using carefully designed tasks in order to develop and validate instructional approaches based on learning theory and targeted for specific categories of human performance. They also are able to conduct externally valid field studies using prototype tutoring systems that are pedagogically based on basic research findings but that teach in real domains. Often the results of

these field studies lead to additional basic research issues to be addressed in the laboratory. For example, the FST Word Problem Solving Tutor (WPS) (see Section 2.4.1 on page 7) was pedagogically based on research findings from TRAIN and other research laboratories around the country. When the tutor was fielded, researchers learned that it was more successful at teaching students to solve concrete algebraic equations (a = 8*5) than abstract algebraic equations (a=b*c). They went back to the TRAIN laboratory to solve this problem and have begun developing a tutoring system that uses a new pedagogy, called *successive abstraction* (based on Reigeluth's elaboration theory). When the approach has been refined until it is optimally effective in the laboratory, it will be added to WPS and revalidated in the field.

Armstrong Laboratory intends for Co-Lab to serve as a national testbed for instructional theory. As such, the result of TRAIN research will influence instructional design at all levels of the educational process. In addition to instructional approaches proposed and developed by AL/HR personnel, learning theorists of nationally recognized merit will be invited to participate in collaborative research efforts.

The TRAIN experiments have already produced a series of important results that are applicable to automated instructional environments in the Air Force. Examples of these results are given in Figure 3 on page 9. Specifically, these experiments provide new and important directions regarding the uses of automated instruction for collaborative learning and in the value of emerging virtual reality technologies to simulation-based training requirements. Groups that have already applied some of these results include the US Air Force Academy, the Air Force, and Ireland's Air Lingus Training Center.

Programmatic Background

This project is performed under an Air Force Office of Scientific Research (AFOSR) directorate grant. This grant approves AFOSR funding from April 1991 to March 1997 of approximately $3.6M.

In-house researchers are supported by a Laboratory Advisory Group consisting of experts from Carnegie-Mellon University, Georgia Institute of Technology, University of Pittsburgh, and Vanderbilt University. Additionally, an extramural program supports researchers from Texas A&M University, Stanford University, and University of Illinois. The Principal Investigator is Dr. Wes Regian from AL/HR.

Planned Products

Short-term products are limited to technical reports, journal publications, and software prototypes. These prototypes will largely be research tools developed to support the study of particular issues. Some, however, can be transitioned into practical use. The console operations tutor, for example, is in use by mission controllers at NASA's Johnson Space Center. A total of eight prototype tutors are currently under development.

Long-term products will include the following:

- CBT-managed group instruction to quadruple training efficiency in high performance tasks (Fortress).
- CBT-managed peer instruction to reduce post-training gender differences in spatial task performance by 70%.
- Direct instruction of expert mental models to reduce post-training error rates by 50% and performance latency by 33% in console operations tasks.
- Student model based active help systems to reduce post-training errors by 50% in a quantitative problem-solving task.
- Cognitively based active pedagogy to enhance quantitative problem-solving performance by 25%.
- VR-based spatial navigation training to produce navigational performance equivalent to real-world navigational training.
- Individual differences in visual attention predict performance over long practice schedules (Fortress).
- Dyadic protocol combines additively with Multiple Emphasis on Components protocol, but Tetradic protocol does not.
- Individual differences in visual attention predict performance over long practice schedules.
- Individual differences in Interaction Anxiety predict optimal match to Individual or Group protocols.
- VR-based console operations training to produce excellent transfer to real-world console performance.

CBT - Computer-based training
VR - Virtual reality

Figure 3. TRAIN Research Findings

- A formal taxonomy of human performance capabilities.
- Technologies for instruction-targeted task analysis.
- Guidelines for developers of automated instruction.

Approach The project seeks to contribute to knowledge for automated instruction in four ways, dividing it into four related tasks.

Task 1: *Taxonomic characterization of human performance capabilities to support generalizations about pedagogy.* This involves categorizing the psychological demands that task performance makes upon people. At any stage in training, focus should be on trainable factors which limit performance. A taxonomy of primitives for describing task performance will provide a clear and concise language for describing task characteristics. This, in turn, will promote and support the synthesis of results across studies. **An initial task performance taxonomy has been drafted and will be published in an appropriate forum. The taxonomy will continue to evolve based on empirical results.**

Task 2: *Development of a set of standard criterion tasks that will allow benchmarked comparisons of instructional approaches.* The intent is to compare the instructional effectiveness of alternative instructional interventions derived from various theories of knowledge and skill acquisition. The criterion tasks will systematically sample across the task taxonomy with the capability to differentially emphasize different psychological demands that task performance makes upon people. They will be designed to be laboratory analogs of a broad spectrum of real-world instructional domains and tasks. **An initial set of criterion tasks has been developed. There are spatial (Space Fortress, Phoenix, Spinner, Maze, Building 578), propositional (Copter, Console, Comprendo, Loader, 911 Dispatcher), and quantitative (Time Bandit, Solver, Spreadsheet) criterion tasks. These are made available to interested researchers free of charge, and a standard set of materials to be provided to such researchers with the tasks is under development. These materials are designed to allow researchers to configure the programs for their experiments and collect data in formats appropriate for their statistical software. Since 1993, qualified researchers have been provided with funding to develop further criterion tasks. Additional criterion tasks continue to be developed.**

Task 3: *Development of automated instructional systems, including ITS, based on formal theories of knowledge and skill acquisition.* In-house work will look at latency, accuracy, and secondary resource load measure performance on criterion tasks as outcome measures. Automated instruction for criterion tasks will be developed. To gain the benefits of widespread use of the criterion tasks, extramural funding will be provided to qualified researchers who are willing to focus their instructional interventions on one or more of the tasks.

Task 4: *Evaluation of these instructional systems in a controlled laboratory setting.* The Co-Lab has the resources to support 50,000 hours of subject-data collection per year. Subjects are retained for 3 hours to 3 months, with sample sizes ranging from 30 to 500 people for each study. This approach allows the specification of target-population characteristics for studies, and then hiring of subjects who conform to

the specified characteristics. **FY1991 funding went to buy equipment and software for the laboratory. Actual studies began in 1992.**

2.4.2 Systems Under Development

This section provides brief descriptions of the tutors developed by the TRAIN effort to serve as research vehicles.

2.4.2.1 Mathematics Abstract Reasoning Tutoring Architecture (Martha)

Martha is an interactive learning environment for algebra word problem solving at the ninth grade level that focuses on the distinction between abstract and concrete equations. Abstract equations are symbolic expressions (A+BX=Y) and concrete equations are numeric expressions (4+(3*5)=19). Students will be taught to generate the concrete expression for solving a word problem, and then to convert this to an abstract expression. Subsequently, they will be taught to directly generate the abstract expression. Students begin by interacting with computer-based training that uses morphing as a metaphor for moving back and forth between abstract and various concrete representations. The computer-based training then uses a drag-and-drop interface to help them perform the following tasks: (1) generate a concrete expression from a word problem and then an abstract expression from the concrete expression, (2) generate an abstract expression from a word problem, and (3) apply the same abstract expression to a different word problem. Students then move into an interactive problem solving environment where they work through a series of problems applying these techniques.

2.4.2.2 Puzzler

Puzzler comprises a set of toolkits for building science experiments and associated simulations to support problem solving. Activities lead students to actively reason about key science concepts. Students solve puzzles to help them induce principles in the concept areas. Concept areas might include plant anatomy (for example, adaptation to environment), photosynthesis, simple mechanics, light refraction, and aerodynamics. Students begin by trying to solve a puzzle (for example, "Use the plant design toolkit to create a plant that can survive in a specific environment"). They are then given relevant instruction with diagnostic remediation. Again they try to solve the puzzle, during and after which they are given expert-based feedback.

2.4.2.3 Solver

Solver is an interactive learning environment (called WPS when fielded in the FST project) for teaching ninth-grade word problem solving. The goal of the tutor is to teach students to analyze word problem statements in order to perform the following: (1) understand the goal of the problem, (2) discriminate between relevant and irrelevant information in the problem statement, (3) write a correct equation to accomplish the goal, and (4) specify the answer in appropriate units. The tutor does not teach algebraic symbol manipulation or mathematical calculation, but is designed to operate in concert with traditional instruction on both. The curriculum is topically modularized, with nine independent modules covering the following topics:

- Algebraic equations.
- Area of geometric shapes.
- Equations in geometry.
- Number sequences.
- Percent and percentages.
- Proportions.
- Ratios.
- Units of measure.
- Volume of geometric solids.

These modules are intended to serve as computer-laboratory exercises to be used in conjunction with traditional lecture-style instruction on related topics. The basic concept is to follow lecture with relevant practice by allowing all students to electronically "go to the board," develop and exercise skills, and receive (automated) individualized feedback to support and optimize skill development. In addition, the modules review the material that has been covered in lecture with a short, interesting, animated graphical sequence. Teachers are able to select modules that are topically relevant to the ongoing lecture curriculum. The pedagogy embodied in the tutor is based on principles of active problem solving (Anderson), elaboration theory (Reigeluth's), categorization by prototype (Rosch), mastery learning (Anderson, Bloom, Shute), and worked examples (Reed, Sweller).

2.4.2.4 Spreadsheet

This model-tracing ITS teaches students how to efficiently solve complex problems using a spreadsheet. The tutor has been validated by comparing its effectiveness to standard on-line help systems and to a tutoring system without a student model.

2.4.2.5 Virtual Interactive System for Training Applications (VISTA)

VISTA is a virtual reality (VR)-based training laboratory for conducting research on the use of VR as an interface to support task performance and training. VISTA training experiments have been conducted for console operations tasks and large-scale spatial navigation tasks.

2.4.2.6 Loader

This laboratory research tool has been developed to assess instructional strategies in the acquisition of console-operation skill. The task simulated by the Loader environment is the operation of a remote crane control arm to load various canisters from a set of storage bins to one or more railroad cars. To provide instructional guidance capable of determining when to intervene and what to say, an expert system was embedded into the delivery system of Loader. The expert system evaluates error criticality in order to determine points of intervention. In addition, the expert system analyzes solution methods to best select the goal structure message. Figure 4 on page 14 provides an example of the Loader Interface.

2.4.2.7 Console Operations Tutor (COPTER)

COPTER teaches real-world tasks performed by NASA personnel during shuttle missions. The tutor embodies specific pedagogical principles, derived from cognitive automaticity theory, to test their effectiveness. COPTER proceeds from declarative instruction, through proceduralization, to automaticity training, and applies different instructional heuristics at each level of training.

In an evaluation study of COPTER, two groups of subjects were tested. One group was trained by COPTER until their accuracy and speed on the tasks were equivalent to an expert (four hours). A second group was trained by COPTER until they were not only as fast and accurate as an expert, but were also able to perform a secondary task while maintaining speed and accuracy (five hours). Subsequently, both groups were tested on a transfer task immediately following training, and after delays of two weeks, one month, and two months. Transfer-task testing was accomplished on a full-scale mockup of the actual console. Speed, accuracy, and susceptibility to negative interference were assessed and com-

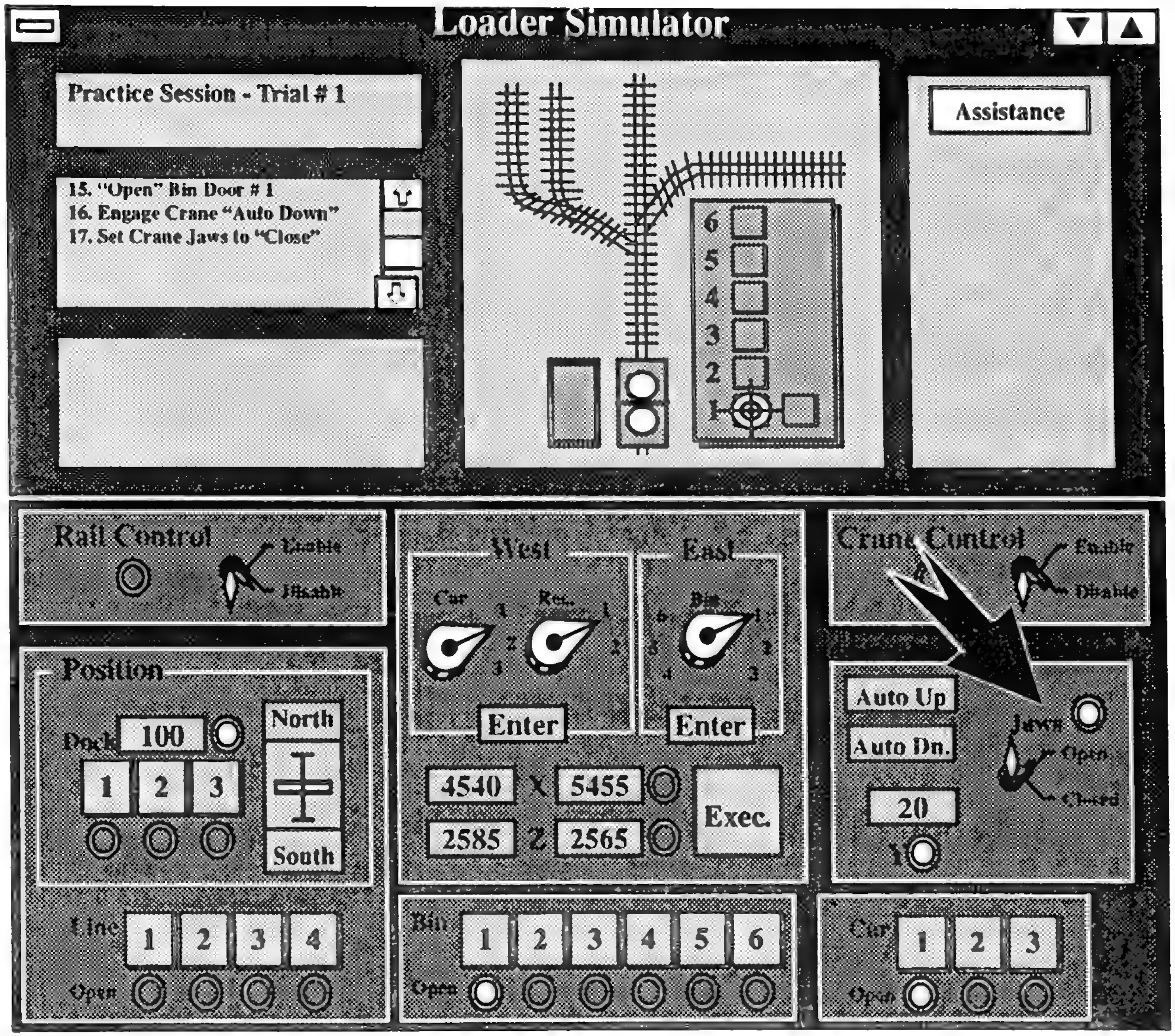


Figure 4. Sample Screen from Loader

pared between the two groups for the various time delays. Subjects trained to automaticity demonstrated transfer-task performance that was significantly faster, more reliable, less susceptible to stress, and less susceptible to skill degradation than was task performance which was not cognitively automated.

2.4.2.8 Phoenix

Phoenix is a tutoring system that teaches cognitive skills associated with performing an instrument-only landing in a fighter airplane. Phoenix is an experimental application of AI to training in a class of task domains which are referred to as high-performance tasks [Regian 1988]. In high-performance tasks, there is more of a requirement for speeded, reliable, and automatic task performance than is found in the typical knowledge-rich ITS domains (for example, medical diagnosis and electronic troubleshooting).

The prototype system trains students to land a simulated aircraft (F-16) using instruments only. During the process, an intelligent coach monitors the student and provides guidance just as an instructor pilot might guide a student pilot. This guidance is presented verbally, using a speech synthesis system to simulate human speech. The student also receives instruction from a simulated Air Traffic Controller, again via speech synthesis but using a different voice from the coach. During early training sessions, the coach has the ability to freeze the simulation to give guidance. The coach also debriefs the student after each training session and prebriefs the student before each training session. During the post-flight debriefs, the coach reviews the student's performance in comparison to performance on earlier flights and highlights specific areas to be worked on in future flights. During the preflight briefs, the coach reminds the student of problem areas that were identified in earlier flights.

Figure 5 on page 16 provides a sample interface screen produced by Phoenix.

2.4.3 Fundamental Skills Training (FST) Project

The overall goal of the FST project is to research, develop, evaluate, and transfer ITS technology to public schools and industry under Federal technology transfer guidelines. To this end, technologies will be developed that are suitable for teaching fundamental skills that can be used to improve reading, writing, arithmetic, and reasoning development through the educational systems. Technologies for remediating fundamental skill deficiencies in recruits (those fundamental skills that are generally applicable) and job holders (more job-specific fundamental skills) will also be investigated, providing insight into the effectiveness of intelligent tutoring technology for fundamental skills remediation.

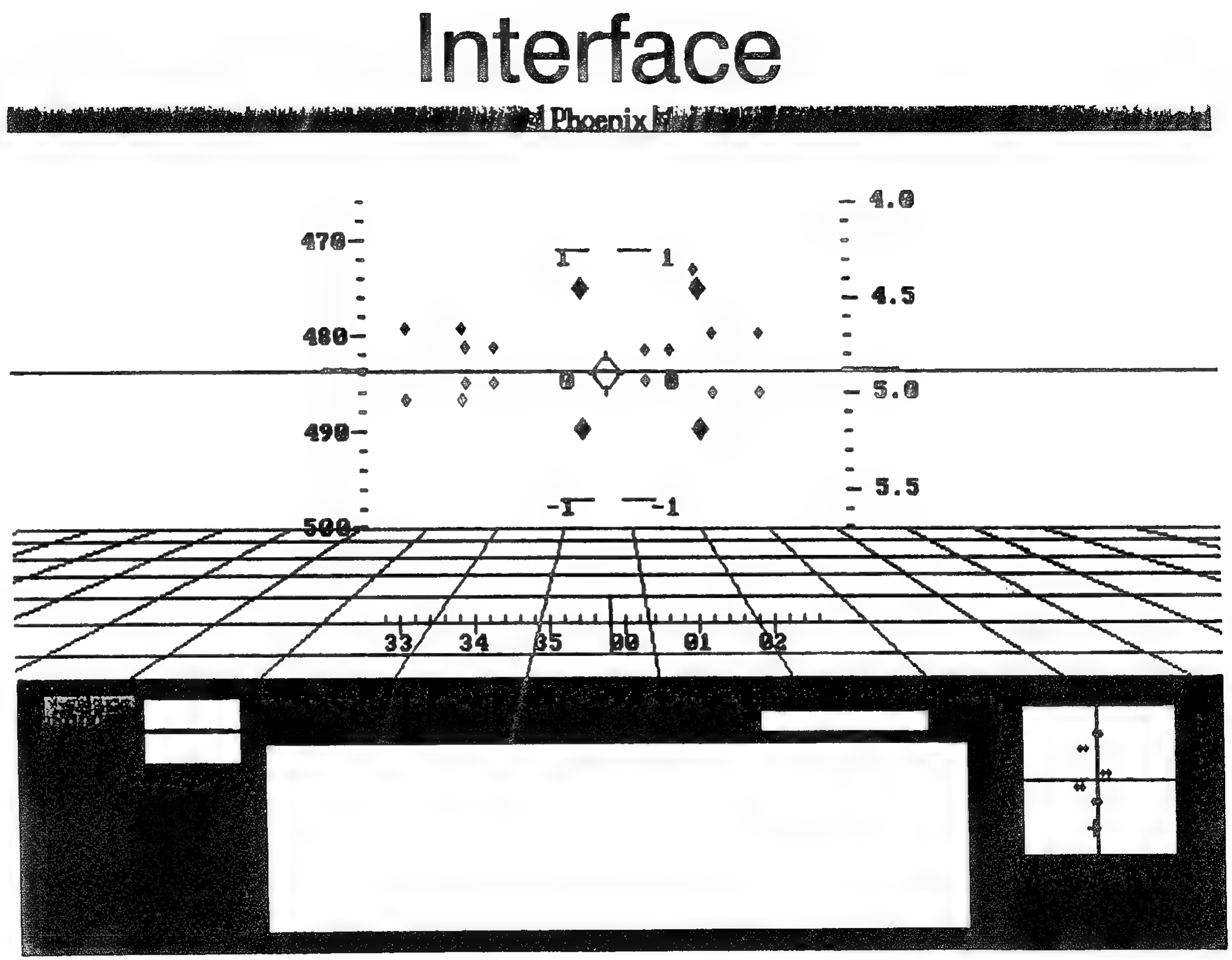


Figure 5. Sample Screen from Phoenix

The combined resources of the DOD, industry, and education allow transfer of new technologies to users outside of DOD (private industry and education). Broad application could increase national education baseline, future recruit capabilities, and US economic competitiveness. This project sets the Air Force standard for educational technology transfer. The strategy is to copyright FST technology and license this software with multiple commercial partners via Cooperative Research and Development Agreement (CRADA).

Programmatic Background

Begun June 1990, the estimated completion date for this project is April 1996. FST is a 6.2 project, with funding provided by Armstrong Laboratory and the other Air Force super labs (under a Memorandum of Understanding) of an approximate total of $4.5M.

As part of the overall approach, this project is being conducted through a CRADA with the University of Texas at San Antonio. The following FST test and evaluation sites have been established through agreements with Armstrong Laboratory, Phillips Laboratory, Rome Laboratory, and Wright Laboratory:

- Dayton, OH: Trotwood-Madison High School, Dunbar High School.
- Albuquerque, NM: Bernalillo High School, Los Lunas High School.
- Rome, NY: Staley Middle School, Mohawk Community College.
- San Antonio, TX: Sam Houston High School, MacArthur High School (pilot test site), and Burger King Academy.
- Allentown, PA: Salisbury Middle School.

Researchers from Wright State University, University of Pittsburgh, Rose-Hulman Institute of Technology, and University of California are also participating in this effort.

Planned Products

The major products will be production prototype ITS for remediation of critical thinking skills in pre-algebra mathematics, reading/writing, and basic science.

Approach

The initial approach to the fundamental skills effort will be the development of specific Air Force personnel training technology, building ITS in mathematics, science, and English; and testing in-house developed tutors to determine effectiveness in remediating fundamental skills deficiencies. A secondary approach will be the development of training technologies for use in public educational systems to improve generic fundamental skills development.

Status

The first tutoring system developed is the pre-algebra WPS tutor. This tutor teaches individuals how to solve specific word problems, as well as more general problem-solving strategies with broader application. The initial instructional system development and knowledge engineering began in July 1990. A pilot evaluation of the system, involving over 350 ninth graders, was conducted from September 1991 through May 1992. A revised prototype was then evaluated in a large-scale field study involving six secondary schools from September 1992 through May 1993. The initial test results were positive, with the tutor increasing student word problem solving performance by about 70%. The most recent version of the word prob-

lem solving tutor was evaluated in three secondary schools during the period September 1993 to May 1994. Figure 6 on page 19 shows a sample screen produced by WPS.

The initial design and development of the Reading-Writing in a Supportive Environment (R-WISE) tutor began in November 1991. A pilot study was conducted to evaluate the prototype, using test students from one San Antonio high school and control subjects from another, from September 1992 to May 1993. A large-scale field study of the enhanced prototype was field-tested at eight secondary schools from September 1993 through May 1994.

The initial design and development of the science tutor began in January 1993. A systems requirement document and a functional specifications document for the life science tutor have been drafted for FST team review. Prototype software development begun. The initial evaluation will be conducted in a pilot study on one high school beginning in September 1994.

Licensing

A Business Plan that provides recommendations and an action plan for technology transfer of the WPS Tutor 4.0 has been prepared. This plan can be generalized to other FST tutors/authoring shells. An announcement was published in the *Commerce Business Daily* soliciting qualification statements from potential commercial partners. In February 1994, an FST Technology Transfer Symposium was held with prospective licensees. One or more companies will be selected to sign a CRADA to transfer the tutors from Armstrong Laboratory to the commercial sector.

2.4.4 Systems Under Development

This section briefly discussed the R-WISE and ISIS tutors. Further information on the first tutor, WPS, was not available.

2.4.4.1 Reading/Writing (R-WISE) Tutor

The second tutor facilitates the development of basic writing skills. The tutor guides students through prewriting, drafting, and revision phases of the writing process, using modules that will structure the students' writing activities and allow them to develop the cognitive skills used in writing.

R-WISE consists of a suite of computerized "tools" to aid ninth graders in learning the art of prose composition and reading comprehension. Four separate adaptive tutors make up the total environment:

- Reading Comprehension. This tutor—based on a graphical representation of complex mental operations—helps the student to draw inferences and to elaborate on the basic concepts being developed in a piece of text. The tool helps to bootstrap comprehension by modeling a process for (1) extracting meaning through concept mapping, (2) formulating inferences, interpretations, and rein-

WPS Interface

FST Word Problem Solving Tutor version 3.1

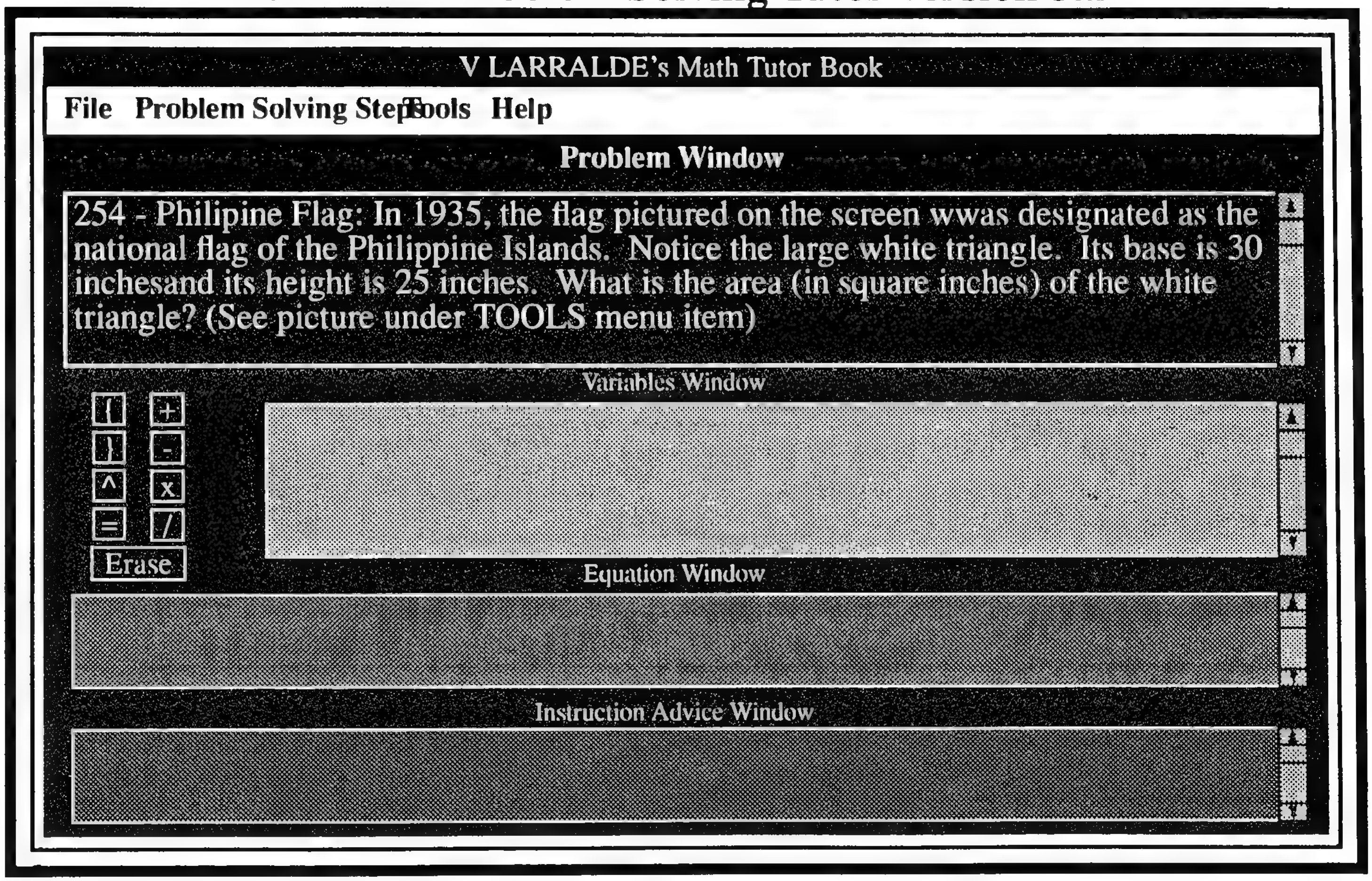


Figure 6. Sample Screen from WPS

terpretations, and (3) engaging stored knowledge and connecting it with the current situation.

- Prewriting. This tutor helps the student generate content while (1) accommodating deficiencies and thereby reducing frustration for a weak writer, (2) enriching the environment and thereby sustaining motivation, and (3) prompting for goal-directed brainstorming.
- Drafting. This tutor mediates a major cognitive shift in the writing process—moving from free-formed ideas to the more rigorous structures of socially acceptable, connected prose (for example, sentences, paragraphs, and whole-text units). By foregrounding activities at appropriate times and relegating others to less prominence, the tool teaches the student to manage the cognitive load of composition.
- Revision. This tutor helps the student to "re-see" a completed draft. Editing, as fostered in this tool, refers to substantial changes, such as improving the style, adding to or subtracting from the content, rearranging parts, or completely rewriting. These more global, deep-structured editing acts are associated with higher-order cognitive skills (for example, discerning patterns in bodies of information, exercising judgement, analysis, and synthesis).

R-WISE uses a hybrid paradigm for interactive instruction. Part of the guidance comes from adaptive tutoring using traditional artificial intelligent (AI) formalisms, and part of the teaching comes from the powers of *reification* (or representing complex processes as manipulative objects on the computer screen). Each tutor acts as a cognitive partner for the student. The computer serves as a mentor, sequencing activities and offering help when asked. All four tutors interact with the student at three different levels of instruction. First, they provide a check list for diagnosing a problem and selecting a fix. Second, they provide adaptive advice in the next-best-move, given the current conditions. Third, they offer an embedded, basic explanation of the specific reading or writing skill the student is working with.

2.4.4.2 Instruction in Scientific Inquiry Skills Tutor

The third tutoring system is designed to teach students scientific inquiry skills in the context of introductory biology for ninth-grade students.

A goal of science education is to produce students who are competent in science. By "competent in science," the project team means that each student will be literate, functional, and critical in the domain of science. To be literate denotes the abilities to obtain, comprehend, and communicate scientific material. To be functional refers to the ability to use the methods, principles, and technologies that pertain to science. To be critical indicates the abilities to assess the soundness of scientific approaches and outcomes and to judge the significance of science and technology in society.

The over-arching goal of the Instruction in Scientific Inquiry Skills (ISIS) tutor is to increase the level of scientific functioning of high school students enrolled in introductory biology. Specifically, students are required to conduct experimental inquiries in the context of a simulated ecosystem. While some of the activities required in conducting scientific inquiry are automated within ISIS, other skills constitute what the students will learn by interacting with the computer. The following are the specific, measurable objectives of this FST life science tutor. Specifically, students will be able to perform the following:

- State a testable hypothesis.
- Design a controlled experiment to test that hypothesis.
- Conduct the experiment in a simulated environment.
- State a conclusion from the experiment.
- Accept or reject the hypothesis.

2.4.5 Intelligent Computer-Assisted Training Testbeds (ICATT) Project

The goal of the ICATT project is to create, demonstrate, and evaluate a capability to rapidly develop and deliver intelligent simulation-based training systems for equipment-related tasks. The program capitalizes on advances in AI and equipment simulation technology to address training for complex equipment maintenance and operation. Both equipment maintenance and operation are critical to Air Force operational commands because success in modern warfighting scenarios depends on effective use and maintenance of sophisticated equipment.

Though instructional effectiveness of ITS is generally not in question, it has been historically difficult to develop such advanced training technology without prohibitive development and hardware constraints. Typical current day ITS development efforts take three years and incur costs around $1M. ICATT authoring shells are intended to allow instructors with no programming experience to perform the following tasks:

- Rapidly simulate any complex device which includes electrical, mechanical, or hydraulic components.
- Use simulated devices as a basis for developing automated instruction.
- Reduce tutor development time by up to 80% (from three years to six months).
- Reduce tutor development cost by up to 95% (from $1M to $50K).

Through ICATT, researchers have developed the capability to rapidly simulate the behavior and appearance of any complex device that consists of electrical, electronic, mechanical, or hydraulic components. As examples, software simulations have been developed for applications as diverse as internal combustion engines, helicopter bladefold systems, and satellite communications consoles. More importantly, researchers have built cost-effective, prototype training systems which use these simulations as a context for training. Additionally, they have developed authoring tools that allow instructional designers to rapidly build ITS using the device simulation as an instructional context. The ICATT program has brought this intelligent simulation-based training capability to a level of cost-effectiveness never before possible on a large scale.

Established customers include Air Education and Training Command (AETC), Air Force Space Command (AFSPACECOM), Wilford Hall Air Force Medical Center, Air Force Special Operations Command (AFSOC), and Naval Sea Systems Command (Mare Island Naval Shipyard). Potential customers include all DOD/Federal agencies, civilian industry, and educational institutions. Technology transfer to the private sector is in progress through a CRADA undertaken with Galaxy Scientific Corporation.

Programmatic Background This 6.3 project is funded at a level of $2.6M for in-house work and an additional $2.7M for contract work. Contract work on the Rapid ITS Development System (RIDES) is being performed by Behavioral Technologies Laboratory at the University of Southern California. Contract work on STEP Writer is being performed by Galaxy Scientific Corporation. Navy and Air Force testbed sites have been identified. A commercial testbed site is still being sought. The Principal Investigator is Mr. Jim Fleming from AL/HR.

Planned Products Major products will be the Rapid Prototype ITS Development System (RAPIDS) ITS Authoring Shell research tool, prototype MITT Writer Authoring Shell, prototype RIDES ITS Authoring Shell, and prototype tutors developed with RIDES for AFSPACECOM and ATC. Products include full documentation, specifications, and ITS developers' guidelines.

Approach Authoring shells are developed in an iterative manner. First, a review of the state of the art is conducted. This provides findings from prior basic and applied research efforts that define the basis for the functionality necessary and possible in

the new authoring shell. The review analyzes a variety of technologies that could contribute to the shell, including instructional strategies, AI, human-machine interaction, computer hardware, simulation software, and computer input devices. Results from this review are used to write draft specifications for the alpha version of the prototype product. These are functional specifications designed to provide a blueprint for defining not so much how the authoring shell will be programmed as what functions it must provide. The alpha version of the shell is then developed. Once completed, it is tested in two phases. First, it is used to build prototype ITS in the laboratory. Then, it is used to build proof-of-concept ITS in operationally realistic environments. Laboratory and operational prototypes are evaluated, and the design-development-prototyping-evaluation cycle is repeated; the evaluation results are used to revise the design specifications, the beta version of the authoring shell is produced, prototype tutors are built, they are evaluated, and the specifications are revised. After specifications of the beta version are revised for the last time, the technology is ready for transitioning to a Systems Program Office, a customer, or other research projects. All products developed during the project are available for transition. This includes the authoring shell, prototype tutors, specifications, user manuals, programmer manuals, and technical reports.

Status Authoring needs have been identified in the limited area of operation and maintenance of complex equipment to support training for AF 2000 high technology career fields. The Phase I ICATT software, termed RAPIDS and intended for laboratory evaluation of instructional approaches to simulation-based training, has been completed. The Phase II ICATT software, termed RIDES Version B1, has also been completed and was delivered to the Air Force in early 1993.

2.4.6 Systems Under Development

RAPID was developed solely for use as a research tool. It is no longer in use and not discussed here.

2.4.6.1 MITT Writer Authoring Shell

The MITT Writer Authoring Shell can be used by instructional developers with little or no programming experience to deliver ITS for training tasks that involve troubleshooting complex devices. A tutor developed with MITT Writer first presents a troubleshooting problem to a student, then monitors that student's actions as he diagnoses the system, compares the student's actions to an expert's actions, and delivers suitable, relevant feedback to assist the student.

Development Status MITT Writer was transitioned to Human Systems Center, System Program Office, in December 1992.

Operating Environment MITT Writer runs on 80386/486-based machines and 1MB RAM. Written in C, using Windows environment.

Evaluation Status Formal controlled tests looking at instructional effectiveness have been conducted.

Developed Tutors Prototype MITT tutors have been developed for the following devices and situations:

- GTCP-85 Auxiliary Power Unit
- C-5B High Frequency Radio
- C-5B Ultra-High Frequency Radio
- C-130 Interphone System
- UGC-141 Teletypewriter Unit
- KG-84 Cryptographic Unit
- US Naval 150-ton Shipborne Air Conditioning Unit
- Acute Care Liver Transplant
- Airborne Warning and Control System (AWACS) Radar Maintenance

2.4.6.2 Rapid ITS Development System (RIDES)

RIDES is an authoring tool for the production and delivery of ITS about devices and domains that can benefit from learner interaction with complex computer-based graphical models. In addition to the experiential learning that is inherent in interactive simulations, the RIDES environment supports the development of highly structured exercises to further promote learning and understanding of the domain. RIDES is designed to support the development of content-based representations of interactive system behavior. This representation is used to automatically select appropriate presentations for a student, based on the student's progress or behavior.

The RIDES authoring approach attempts to significantly improve the productivity of instructional developers both in model development and authoring of a variety of pedagogical exercises to provide instruction based on the resulting models. Using a set of integrated editors in the RIDES environment, tutor development consists of authoring "objects" and their behaviors, authoring "knowledge units" that contain information about the modeled domain, and building "instructional units" that exploit the authored models and knowledge. Instruction authoring tools support a wide range of learning activities (for example, nomenclature, identification of system components, relationships among components, operations, procedures, interpretation of system indicators, and performing fault diagnosis). When completed, RIDES will operate in a networked environment where an instructor can communicate with any of several student consoles.

RIDES will be used to generate approximately five ITS. Research will then be performed on these to determine the most effective designs for ITS for simulation-based training.

Development Status

RIDES Version B1 is operational. RIDES has been used to develop two preliminary prototype tutors in AFSPACECOM domains. These tutors will be used to evaluate RIDES functionality and interfaces. Development of a tutor to teach instructors how to use RIDES has started. This tutor is being built with RIDES.

RIDES Authoring

Figure 7 on page 26 provides an overview of one process for building a learning environment using the set of RIDES fully integrated editors. In this approach, the author begins with a set of learning objectives that are the goal of the learning environment that is to be built.

Using the RIDES objectives editor, the author enters the learning objectives (and the prerequisite relationships that hold among these objectives) into a normative model of the student in the target domain. A preliminary design for the instructional units that will be used to attain the objectives is prepared using the RIDES instructional unit editor. Then the author can begin building the interactive graphical model of the domain of interest. This process is carried out using a graphical scene editor and a number of editors that let the behavior of graphical objects be specified. Typically, this is a highly interactive process in which objects are drawn, assigned behaviors, tested, and modified repeatedly until the model looks and behaves as desired. The author may then create browsable knowledge units, which can be used to provide on-line, content-based help to students, and which can also be used to automatically generate certain types of instruction. In many cases, knowledge unit authoring will be carried out in tandem with model authoring. The author finishes by building the instructional units that were partially specified earlier in the process. This can be done either by generating patterned exercises or by custom-authoring the required instruction. As each instructional unit is built, it can be tested in the authoring environment. The complete course can then be tested in its entirety, revised as necessary using the editors, and fielded for further testing and application.

The process described here is not the only one by which courses can be developed in RIDES. Less formal methods are frequently appropriate when a small simulation-based lesson is created quickly to serve an immediate need. RIDES permits either type of development approach: one based on a thorough top-down analysis, or a more informal, content-centered approach.

Operating Environment

Currently, RIDES runs on 80484-class personal computers with 8 to 16MB RAM, under Unix. It is written in C++.

Evaluation Status

Preliminary software evaluation and debugging has been completed.

Developed Tutors

Two prototype tutors have been developed with RIDES.

TOE

The Tutor for Orbital Elements (TOE) is intended as an introduction to orbital dynamics. It focuses on familiarizing the students with the classical set of six orbital elements. These elements are sufficient to completely describe the size, shape, and orientation of an orbit at a particular point in time. The tutor depicts these elements with respect to earth satellites and teaches the terminology typically used to describe the orbital paths of earth satellites. Quantitative modeling is used to simulate earth satellite orbits and explain the relationship to the set of elements and

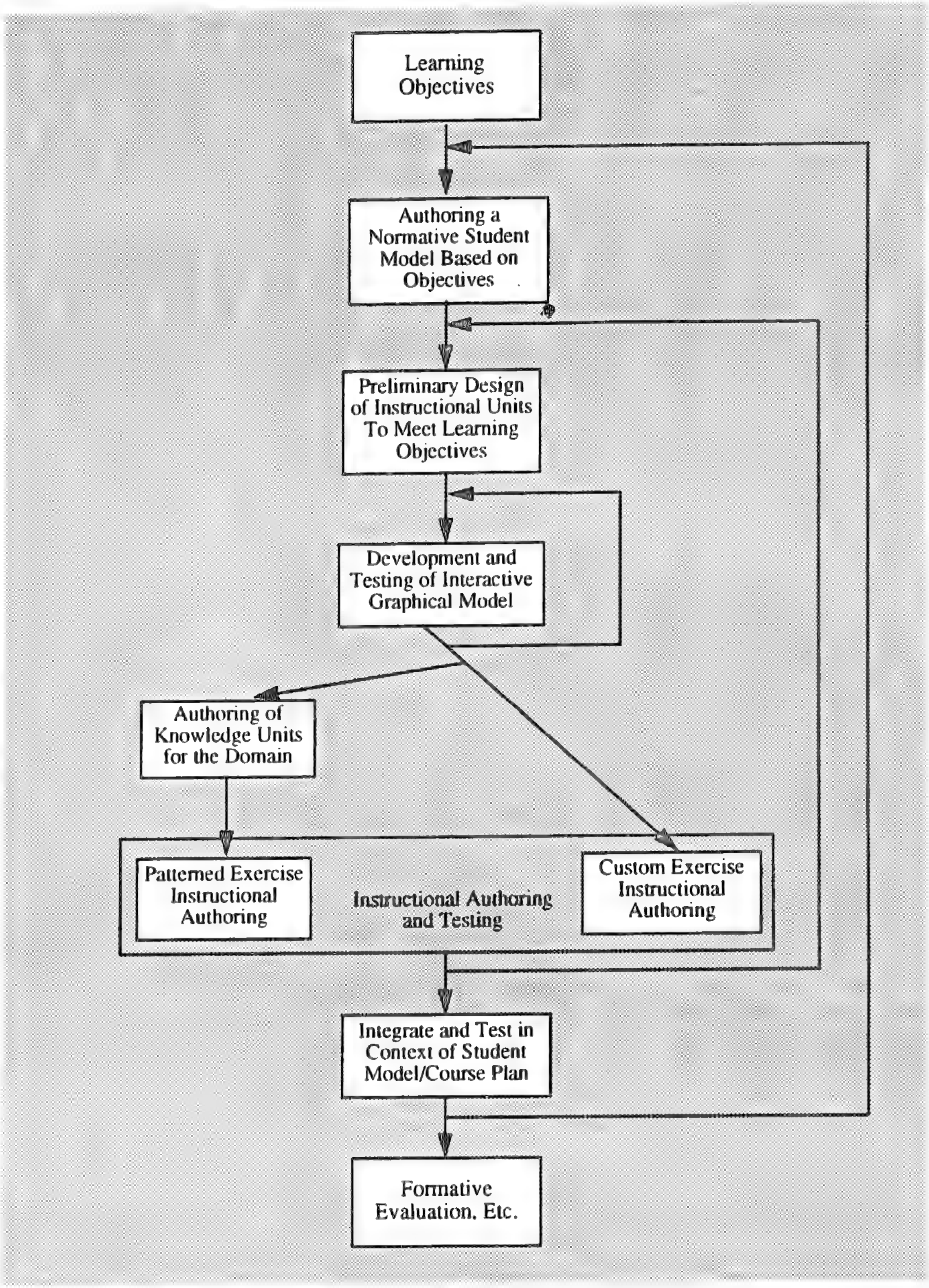


Figure 7. A Process for Applying RIDES to Course Development

the resulting orbital path. The tutor uses an interactive learning environment to provide both guided instruction (which allows for incremental elaboration of the orbital elements and their dynamics) and discovery learning.

TOD The Tactical Operations Display (TOD) tutor is designed to teach students how to operate the Tactical Operations Display of the Radar Set AN/FPS-85 at the 20th Space Surveillance Squadron of Air Force Space Command. The TOD is used to monitor tactical mission performance; that is, initiate, modify, and curtail radar activity as required to ensure optimum mission performance, and act as a backup means of inputting radar tasking normally received from North American Aerospace Defense (NORAD) Space Surveillance Center (SSC) by data link. The tutor uses a basic simulation of the TOD console and keys to present instruction that teaches a student the names and functions of console parts and how to operate and configure the TOD based on a set of commands given by the tutor.

2.4.6.3 STEP Writer Authoring Environment

In response to customer requests, Armstrong Laboratory is producing a new and innovative tutorial production tool called Step Writer. The Step Writer authoring environment supports the production of intelligent tutors which require the student to step through some procedure or activity. Basically, this software allows a non-programmer who is an expert on some specified field to produce a tutorial which teaches procedural skills.

Unlike traditional computer-based training, the resulting tutoring contains elements of intelligent computer-aided instruction. It assists the student in the learning process by providing on-line, timely, and meaningful feedback which relates directly to the student's current situation or activity in the tutorial session. It does this by offering procedural guidance aimed at correcting deficiencies in the student's understanding of the procedures involved in the target domain.

Basically, Step Writer provides training support for three major functional areas: on-line checklists, sequential actions, and assembly and disassembly of parts. Checklists are simply on-line representations of what is now done on paper. An example of sequential actions would be to first plug is a motor power supply, then turn on the power properly. Each action must be performed in sequence in order for the next action to take place. Assembly or disassembly of parts concerns the matter of constructing or dismantling an item of equipment. Examples include an M-16 weapon, a gear box, or a timing mechanism which must be assembled or disassembled in a specific sequence. In all cases, the software would enure accuracy of performance, based on parameters set by the expert who authored the system.

Development Status

This software has been designed to support the needs of the US Air Force and the DOD. It grew out of requests from Air Education and Training Command units who felt a gap in the area of procedural on-line training capabilities. Specifications for Step Writer Version 1.0 have been completed. The product is scheduled for completion by the end of September 1994.

Operating Environment

Step Writer operates on 80386/486-based or higher, IBM-compatible platform with VGA graphics and Windows 3.1 or higher. It is written in C.

2.4.7 Basic Job Skills (BJS) Project

Technological advances in modern Air Force weapon systems and associated support equipment have impeded the skill development of system maintainers in several major respects. First, the automation of many routine maintenance tasks has removed important learning opportunities for apprentices. Second, increased system complexity, combined with improved system reliability in certain areas, has simultaneously increased the possible sources of malfunction in failed systems while decreasing practice opportunities for personnel to isolate faults and make repairs. As a consequence, the time needed to achieve expert levels of performance has been dramatically extended. Moreover, the cognitive demands on high-tech maintenance technicians are currently being broadened through force restructuring initiatives such as Rivet Workforce. AL/HR has responded to this crucial problem with the large-scale program Basic Job Skills (BJS) Research Program.

The overall effort is designed to meet two goals that are central to the mission of a modern Air Force:

- Accelerate the complex skill development in aircraft maintenance specialties with (1) systematic analyses of human maintenance experts, and (2) work-center-based training that is both tightly coupled to authentic high-tech problem-solving demands and predicated on principles of apprenticeship learning.
- Capitalize on common skill demands within and across advanced weapon systems to increase mental flexibility and skill generality among maintenance personnel.

The approach is guided by theoretical and methodological advances in the cognitive sciences regarding complex human performance and the generality of acquired skills.

To achieve the two-part goal, an integrated analytic-training technology is being developed to specify, assess, and enhance the cognitive components of skill that define high-tech maintenance expertise and provide methods and models of analysis and training

that effectively accelerate skill acquisition and foster skill generality to enhance apprenticeship performance. Tasks will include work in the following areas:

- Advancement and codification of cognitive task analysis methods for representing expert knowledge and abstracting skill and knowledge commonalities across Air Force specialties.
- Development and evaluation of cognitive theory-based diagnostic achievement tests for assessing cognitive skill deficiencies and prescribing, guiding, and evaluating instruction.
- Advancement and codification of tutor development methods based on modern cognitive theories of skill acquisition and principles of apprenticeship learning.
- Development and evaluation of testable models of apprenticeship tutoring systems that address increasingly large families of avionics and mechanical maintenance specialties.
- Long-term validation of BJS cognitive skills training against actual performance on the job.

The technologies shall be build through exploratory and advanced development activities that result in a series of successive approximations of interdependent cognitive task analysis and training development methodologies. Associated with each approximation (or generation) of the methodologies will be testable models of instruction (training systems) that are applicable to increasingly large families of avionics and mechanical specialties.

The instructional approach of cognitive apprenticeship is of particular interest. This approach is based upon the opportunity to experience the most difficult aspects of cognitively intense jobs in a simulated work environment where assistance, in the form of an intelligent computer-based coach, is always available and where there are opportunities to reflect on simulated work experiences. This project has developed two generations of tutors for training a specialized electronics maintenance job in the US Air Force, namely the F-15 manual avionics test station technician speciality. Both generations of the training system, named Sherlock I and Sherlock II, have worked well, in terms of success in fostering high levels of job expertise and, with Sherlock II, promoting transfer to new electronic troubleshooting tasks on novel equipment. Two field tests have shown the system to be highly effective, and it has been deployed by the Air Combat Command since February 1994.

Programmatic Background

This project includes both 6.2 and 6.3 funding. Work began in April 1990 and is funded through December 1995. In-house researchers are supported by work conducted under Air Force contract F33615-90-D-0008. The University of Pittsburgh Learning Research and Development Center (LRDC) is the primary contractor. The Principal Investigator is Dr. Alan Lesgold and the Co-Investigator is Dr. Sandy Katz. There are two subcontractors: Bolt, Beranek & Newman (BBN) and Educational Testing Service (ETS). Total funding amount is approximately $4.5M.

Additionally, experts from the University of Pittsburgh, University of Michigan, and University of Colorado support the project through participation on a Scientific Advisory Board.

Planned Products

In addition to various technical reports, the expected products of this work include the following ITS:

- Avionics Troubleshooting Prototype ITS (Sherlock II)
- F-15 Hydraulics ITS (HY-DRIVE)
- F-15 Comm-Nav Penetration Aids ITS
- F-15 Radar ITS

The four tutors employ basically the same architecture; they target the same conceptual skills, system knowledge, procedures for system operation and maintenance, and the same troubleshooting strategies. They will be built using reusable components of Sherlock.

Approach

The overall approach is depicted in Figure 8 on page 30. The following delivery orders outline the elements of this project.

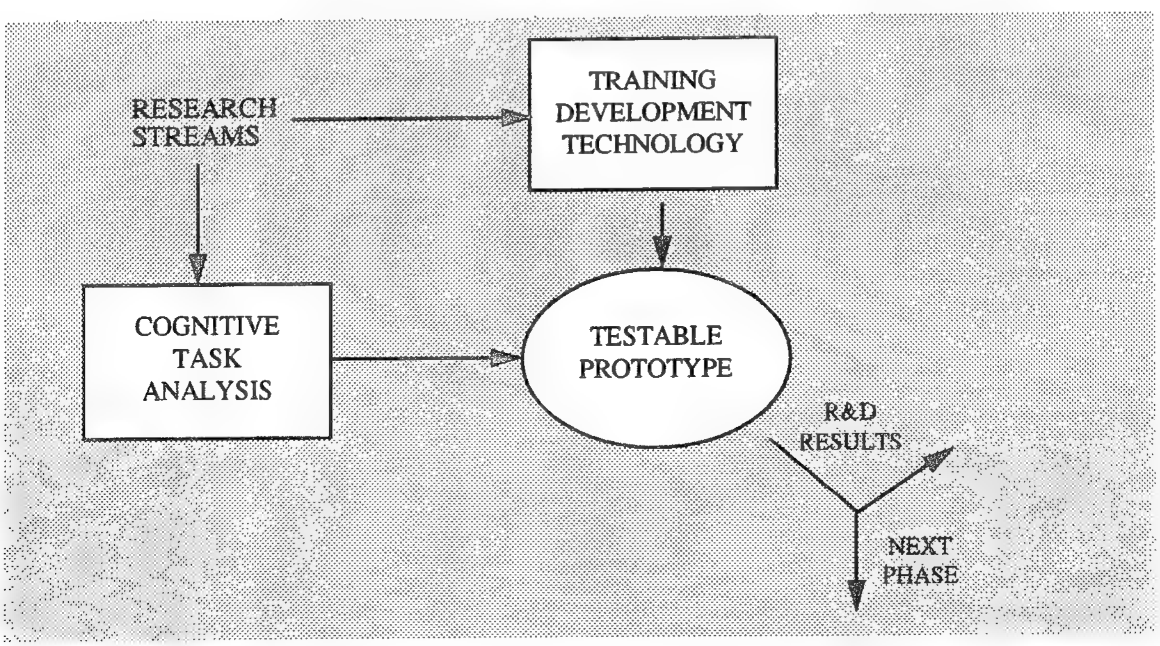


Figure 8. BSJ Approach

Delivery Order 0001: *Cognitive Task Analysis, Student Diagnosis, and Software Design for Troubleshooting Tutoring Systems.* The objective of this research is to develop and implement research plans to build and evaluate avionics and mechanical ITS to teach troubleshooting to fighter aircraft maintenance personnel. This effort shall include conducting cognitive task analyses for a Mechanical Job Family Tutor (JFT); developing cognitive-theory based diagnostic achievement tests for evaluating Job 1 of an Avionics JFT called Sherlock I; evaluating the instructional effectiveness of the tutor for Job 1 of the Avionics JFT; and proposing a plan for integrating the computer architectures of Sherlock I with other avionics jobs in the family. **This delivery order has been completed.**

Delivery Order 0002: *Instructional Methods for Fostering Adaptive Expertise.* The objective of this research effort is to develop and evaluate the effectiveness of alternative instructional techniques in fostering the acceleration and generality of complex skill acquisition. F-15 avionics maintenance technicians with variable levels of training and experience will be the population of interest. The effort shall include developing a detailed research plan for the conduct of multiple learning studies; developing experimental materials to be used in the studies; piloting and subsequently conducting the proposed experiments; and documenting findings in a technical paper. **The work called for by this delivery order changed to require further implementation of the software in order to complete Sherlock II for field evaluation. That work has been completed.**

Delivery Order 0003: *BJS Job Family Tutor Development.* The purpose of this task is to conduct research activities to advance the instructional development and software design for the BJS JFT. This effort includes (1) completing instructional development activities for the F-15 hydraulics tutor HYDRIVE (Job 1 in the Mechanical JFT), (2) developing HYDRIVE student modeling/diagnosis procedures and pre- and post-evaluation methods/instruments, (3) completing software development for HYDRIVE, (4) proposing plans for integrating the computer architectures of HYDRIVE with other jobs in the Mechanical JFT, (5) integrating software for Job 2 in the Avionics JFT with Job 1 software (in Sherlock II), (6) documenting development of Sherlock II in a procedural guide, and (7) completing software development for the demonstration Avionics JFT called RIVTEK. **This delivery order has been completed.**

Delivery Order 0004: *Enhanced Learning and Assessment Component for BJS Tutors.* The purpose of this order is to conduct research to develop enhanced learning activities, student modeling and assessment procedures, and software tools and models for the BJS tutoring systems. Both single job tutors (SJT) and JFT will be targeted by the research. This effort includes (1) enhancing learning activities in HYDRIVE (the F-15 hydraulics SJT) to include instruction that focuses on the interplay of electrical and mechanical components; (2) applying probabilistic inference networks to student modeling and assessment capabilities in both HYDRIVE and Sherlock II; and (3) documenting selected Sherlock II software tools and generating documented software models for use in RIVTEK. **This delivery order has been completed.**

Delivery Order 0005: *Formative Evaluation and Software Engineering Production Plan for BJS Avionics Tutors*. The purpose of this order is to conduct research to produce a release of Sherlock II for formative evaluation and to generate a software engineering plan to guide the development of multiple avionics tutors in the future. This effort includes (1) selecting and implementing specific tutor components to fulfill explanation, assessment, and interface requirements for a Sherlock II.5E—a release for formative evaluation purposes, (2) conducting a cognitive task analysis of F-15 flightline avionics troubleshooting tasks, (3) using cognitive task analysis results and the Sherlock II.5 software design to generate a software engineering plan for future development efforts where multiple avionics tutors are produced, and (4) conducting a series of reviews of tutor evaluation outcomes. **This delivery order has been completed.**

Delivery Order 0006: *BJS Tutor Evaluation and Advanced Tutor Development Technology*. The purpose of this order is to conduct research to evaluate BJS tutors and advance the integrated BJS assessment-tutor technology for development of JFTs. This effort includes (1) the development of a diagnostic assessment component for HYDRIVE; (2) software revisions to HYDRIVE, based on field trials of learning activities involving the flight control subsystem; (3) refinements of Sherlock II, based on formative evaluations of the tutor's interface, simulation, coaching, and assessment components; (4) development of Sherlock II in Microsoft (MS) Windows; and (5) software simulation for avionics subsystems targeted by Job 1 of the F-15 Flightline Avionics JFT. **Work on this delivery order is continuing.**

Delivery Order 0007: *Development and Field Evaluation of Initial Version F-15 Troubleshooting Tutors*. The purposes of this delivery order are as follows: (1) deploy, evaluate, and support tutorial software for the first "initial version" avionics troubleshooting tutor (Sherlock II) at four ACC wings, and (2) initiate the development of the first F-15 Flightline Avionics "initial version" tutor. As part of the field evaluation of Sherlock II, user orientation and training materials will be developed and a training course will be conducted for ACC training monitors. Further, a formal evaluation plan for Sherlock II will be developed and implemented so that the field experiences with Sherlock II can inform the development of subsequent tutors. In conjunction with the development of the first flightline avionics tutor, the interrogation skills used by experienced technicians during post-flight debriefs with pilots will be studied carefully and potential instructional schemes will be proposed. **Work on this delivery order is continuing.**

Delivery Order 0008: *Trade Study for F-16 Avionics Tutor Development*. The purpose of this delivery order is to produce a trade study analyzing basic design approaches for the development of the F-16 Avionics Tutor in a two-month period of performance. This analysis will consider the options for development, emphasizing cost, schedule, and technical performance considerations. The trade study will show the method for optimal reuse of previously developed tutor components, and will identify areas of overlap between the F-15 Avionics Tutor, the Sherlock II tutor, and the F-16 Avionics Tutor. **This delivery order has been completed.**

Delivery Order 0009: *F-16 Flightline Tutor Development.* The goal of this delivery order is to pursue acceleration of the development and fielding of tutor technology. A tutor for F-16 Flightline Avionics (C-shop) will be developed. This development will be modeled to follow the development of the F-15 avionics tutor to capitalize on the synergistic opportunities presented through these parallel developments. **Work on this delivery order is continuing.**

Delivery Order 0010: *F-15 Flightline Tutor Development.* The purpose of this order is to design, develop, integrate, and test BJS tutors for F-15 flightline jobs, and advance the integrated BJS cognitive task analysis and tutor development methodologies. This effort includes (1) completion and testing of the F-15 Flightline Avionics (C-shop) tutor software, (2) design and development of the F-15 Flightline Avionics A-shop tutor, and (3) conversion of HYDRIVE for delivery on ACC-provided hardware. **Work on this delivery order has just begun. The task has been expanded to look at the provision of expert explanation generation capabilities, and to include studies that will determine how the tutor could be expanded to support collaborative training.**

Potential Follow-On The addition of Delivery Order 11 to the contract is currently under consideration. This new delivery order calls for the development of support materials for HYDRIVE and the F-15 Flightline Avionics (C-shop) tutor, and the provision of software support for Sherlock II. In preparation for the field implementation of HYDRIVE and the C-shop tutors, user orientation and training materials will be developed and a training course will be conducted for ACC training monitors. Further, Delivery Order 11 calls for developing and implementing a formal plan for evaluating tutors so that the field experiences with a tutor can inform the development of subsequent tutors. Further delivery orders may be added to the contract in the next year.

2.4.8 Sherlock ITS

Sherlock trains technicians who work with a test station. This is a tool for diagnosing failures in aircraft parts, specifically navigation electronics from the F-15 tactical fighter plane. It is a giant switch, like a telephone exchange, that connects power sources to various contacts on an aircraft part and also connects measurement devices to other contacts on the part. The aircraft modules comprise a limited set of circuit boards and, generally, technicians have little difficulty learning to use the test station to diagnose these. The problem arises when the test station itself fails or, more specifically, in determining whether a bad reading is caused by the aircraft module or the test station itself. The test station is composed of massive amounts (approximately 70 cubic feet) of discrete electronic components soldered to printed circuit cards and the special tools for diagnosing its failures are minimal. So while the everyday job of these technicians is to use good tools to make simple diagnoses, their occasional job is to use minimal tools to make complex diagnoses. It is this

second, more difficult, chore that Sherlock teaches by making the cognitive processes of test station troubleshooting more observable so that they can be learned and practiced.

Sherlock is a new approach to training specific cognitive skills. As an intelligent coached practice environment, it has much in common with traditional ITS, but is not as much driven by a dynamic student model. Also, its intelligence primarily is used to respond to student requests, rather than to intervene actively. The approach has several distinguishing characteristics:

- The learning activity is centered in a simulated work environment.
- The learning activity is centered around problems that exemplify the hardest parts of the job for which one is being training.
- For each problem, two kinds of activities occur: (1) the student solves the problem, requesting advice from the intelligent tutor as necessary, and (2) the student reviews a record of his problem-solving activity.

The approach shares several instructional components with traditional apprenticeship approaches: observation, coaching, fading of assistance, shared problem solving between teacher and student, and situated learning in the context of subsequent knowledge use. Where cognitive apprenticeship approaches go beyond the traditional approach is in developing ways of facilitating the externalization of expert problem-solving activity and the student's skills of learning. Like expert cognition, expert teaching is often inaccessible to study. Sherlock helps to remedy this problem: expert teaching is specified by rules and is replayable, essentially Sherlock can be viewed as an externalization of expertise. Additionally, Sherlock provides an opportunity to answer questions such as the following:

- How should coaching be provided and when should it be used? The expert teacher depends on intuition. (Whereas Sherlock I used explicit models of the student's competence to drive the coaching and fading of feedback, Sherlock II allows students to select the type of coaching they require.)
- How can coaching be individualized for a student? The expert teacher may deal with many students at once, or at best rely on a general sense of a student's ability. (Sherlock I gave different levels of help according to the student's current state of achievement. Sherlock II allows students to specify the level of help they desire.)

Sherlock can also support investigation of the social environment of learning. In field testing, Sherlock was embedded in a social situation as well as a context of application. In discussing possible future developments of Sherlock, the system could be used to foster such aspects of socially situated learning as peer tutoring and role reversal.

Sherlock II provides a simulation of the work environment using a combination of video and computer graphic displays. Simulated controls can be operated with the computer mouse, and the displays change to reflect an underlying computer simulation of the devices being tested. Since the fundamental activity of troubleshooting in this job is making tests with meters, this is provided realistically by having icons of meter probes that can be "attached" to video images of device test points.

A collection of tools for reflection has been developed for Sherlock II. These can be used during troubleshooting and for reflective follow-up. One tool provides an intelligent replay of the trainee's actions. A trainee can "walk through" the actions he just performed while solving the problem. In addition, he can access information about what can in principle be known about the system, given the actions replayed so far. He can ask what an expert might have done in place of any of his actions, get a critique of his actions, and have his actions evaluated by the system. In addition, extensive conceptual knowledge about the system's function is available from intelligent hypergraphic displays of an expert's circuit model schematic drawing. In these drawings, the boxes that stand for circuit components are all mouse-sensitive and can "tell about themselves." Boxes in the diagram are color coded to indicate what is known about them, given the tests carried out so far. Circuit paths are color coded to indicate whether the electrical properties of those paths are known to be appropriate or inappropriate for the function that has failed. To ensure that the trainee does not substitute looking at labels in the displays for inferring what circuity is involved in the functional failure being diagnosed, components are only labeled when the trainee has identified them. There is also a tool for displaying an expert solution to the problem, again with extensive conceptual information available as appropriate to each step. Further, there is an option for side-by-side listing of an expert solution and the trainee's most recent effort.

Figure 9 on page 37 shows a sample Sherlock screen.

Development Status Sherlock II is in operational use.

Architecture The Sherlock architecture is based on the LDRC Tutor Framework. This framework is implemented as a flexible collection of abstract classes, whose relationships and intercommunication mechanisms are pre-established. The Sherlock ITS

is one particular instantiation of this framework, specializing the electronics diagnosis classes to have the "Sherlock" flavor.

Team Concept

The Tutor uses a "staff" team concept. A teaching staff needs to have domain knowledge and must include a domain expert. It must know how to convey this knowledge and, therefore, must include a coach. It must be able to assess the abilities of the users, and so includes a student judge. The LDRC Tutor Framework has objects that take on each of these roles. From the Tutor's point of view, the staff consists of these individuals—in reality, each individual may be a spokesperson for a team of specialists. For example, a "coach" may be a coaching team, consisting of an expert to make inferences based on the users' actions, a student judge to be able to tailor coaching to the users' level of understanding, and a rhetorician to be able to convey these ideas clearly.

SessionMonitor Object

Team members may base their actions on what has already been done. The SessionMonitor is used to record the user-tutor/tutor-user interactions. The explicit communications channels make it easy to "tap" into a session, and record all occurrences of events, their parameters, and the object which raised it. The "overheard" conversation can be recorded onto a SessionMonitor.

TutorCoordinator Object

The TutorCoordinator serves two roles distinguishable not only by their function, but by when the TutorCoordinator plays each role. The first role, as bootstrap, is used prior to a tutoring session. Here the TutorCoordinator selects and instantiates components to be used. Component selection is configurable according to the user, installation site, and hardware platform requirements (configuration is managed by the TutorConfiguration classes). During component instantiation, the TutorCoordination establishes relationships between the system's components. The primary relationship is the "has-a" relationship, established through the setting of a component's instance variable. Another relationship is the "acquainted-with" relation, established by setting up communications channels between objects.

The second role, as communications liaison, is used during the tutoring session. During a tutoring session, the user may elicit or be directed to engage in a different mode of interaction. Each mode corresponds to a restricted conversation in which the user engages one or more of the Tutor's staff members. The coordination of user-tutor interactions requires the sequencing of system capabilities by the enabling and disabling of particular modes of interaction between the user and the Tutor components, depending on the user's actions. The criteria used to determine when to enable and disable these modes of interaction give the tutor its characteristic "flavor." The TutorCoordinator coordinates inter-tutor component interactions by "broadcasting" user events to other interested parties.

A tutoring session is divided into phases. In each phase, the user may only elicit to engage in a subset of the system's possible modes of interaction. For example, Sherlock presently divides the tutoring session into problem detection, problem correction, and solution analysis phases. Once entered into a new phase, the user may not return to a previous phase. This marching of the user through phases of the solution is a characteristic of the Sherlock specialization, and not a characteristic of its TutorCoordinator ancestry.

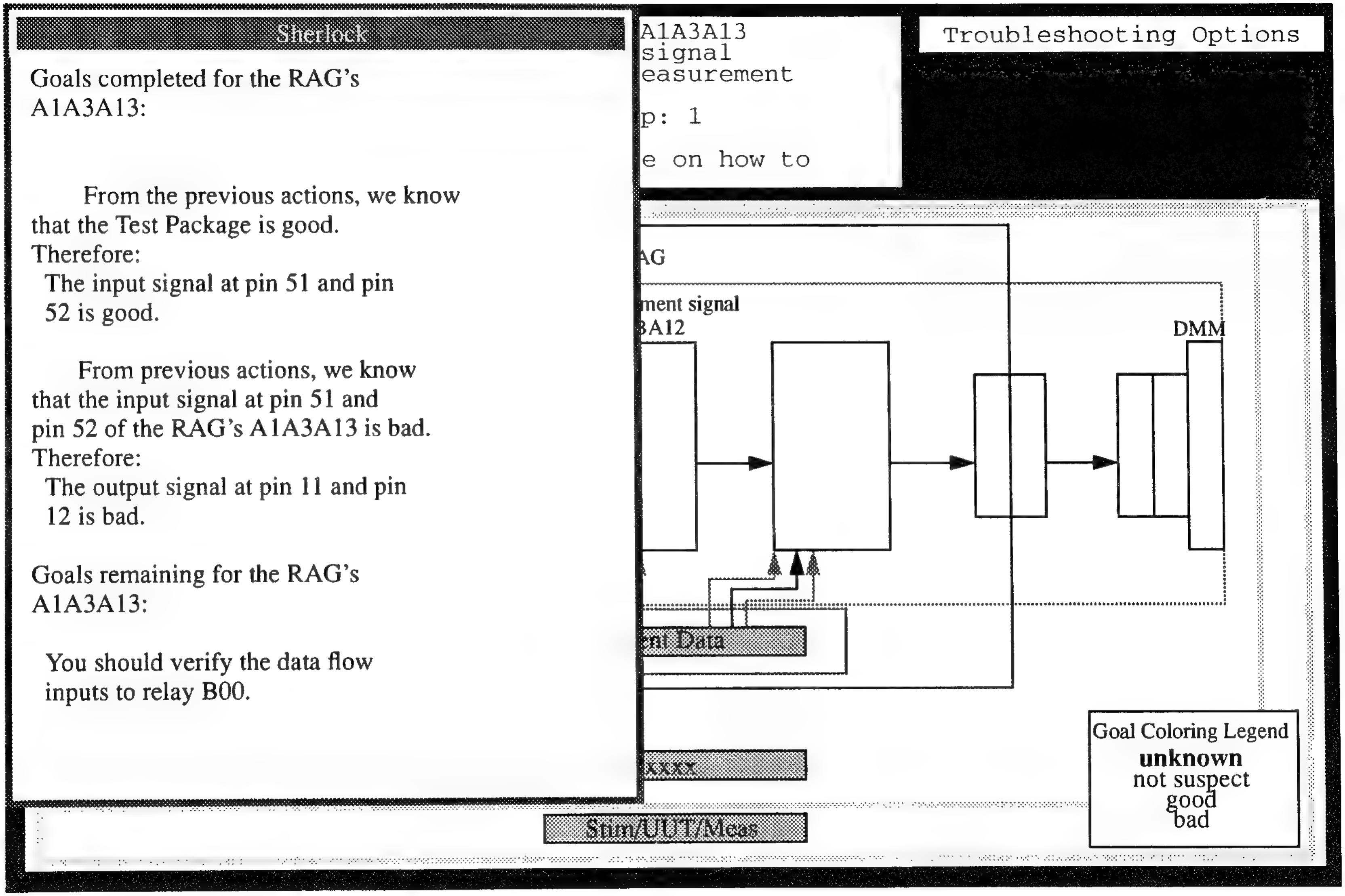


Figure 9. Sample Sherlock Screen

IOManager Object

The IOManager is a resource manager for multiple clients. It makes the appropriate provisions by which the user can engage in conversations with the Tutor's components. Thus, each provision will be some IO module specifically designed for providing the type of restricted interaction which corresponds to a mode of operation. The IOManager knows what IO modules it has at its disposal. It can be envisioned as having a toolbox of graphical user interfaces (GUIs) and, currently, knowledge of the contents of the toolbox is encoded in IOManager specializations. Provisions may be platform specific, or changed according to new GUI theories, independent of the Tutor's structure. They are abstracted away from clients.

Provisions

There are two components to an IO module: the physical component, provided by ApplicationGUI specializations, and the parsing of user actions, provided by ActionParser specializations. By keeping the physical layout separate from the IO module's behavior, the same provision can be given different looks, using the same action parsing.

ActionParser Object

ActionParsers collect the parameters of an action, and when they detect that the action is complete with legal parameters, raise an appropriate event. For example, to make a measurement, a piece of test equipment and a component to test must be selected. The user may select either first, but both must be selected before attempting to select and place a probe. At any time thereafter, the user may change his mind about a probe placement or the test equipment setup. Only once the user has selected both probe placements can he make a measurement. Although some actions have prerequisites, others may occur in any order. Thus, an ActionParser is like a finite state automaton, accepting a wide range of user interactions.

Communications Manager Object

The relationship between an IO module and IOManager client is like the "acquaintance" relationship. Here, two components have agreed to work together as a team, though neither is a part of the other. The nature of this relationship requires that someone provide a means of communication between the team members.

SessionPlayer Object

A SessionPlayer object is able to reply events recorded on a SessionMonitor by asking objects assuming the role of the original event raisers to raise them again. By recording events rather than message sends, the Tutor can replay old session on different tutor configurations, so long as the provisions extended by the IOManager live up to the old contractual agreements of what events to pick up and raise. Only a mapping between the event originators and their new impostors is needed.

Domain Knowledge

Sherlock's basic mode of interaction with trainees is to pose realistic troubleshooting problems and to provide expert advice on demand. Accordingly, the knowledge in Sherlock is organized around problems, problem solution methods, and the constraints imposed on problem solution by the structure of the work environment. Much of Sherlock's domain knowledge is represented in a separate structure specialized to each problem that it presents. Problem-based knowledge organization is efficient because it is based on what is called the "effective problem space" where only those moves (actions that move the solution process from one partial solution state to another) likely to be made by experts or trainees, plus additional moves that are infrequent but consistent with the troubleshooting approaches being taught, are considered. To provide further structure, the various pieces of

problem space paths are grouped into a hierarchy of larger units called subgoals; each subgoal is a piece of problem-solving procedure. Some are strategic, some tactical. Some involve heuristic plans, others involve fixed procedures. The actual Sherlock knowledge structure for a problem is a network in which the nodes are these subgoals rather than the more microscopic actions that they summarize. That is, most subgoals subsume a lot of mental activity; they are abstracted problem subgoal structures.

Evaluation Status

The most recent testing was conducted in Fall 1993. Done with a criterion of real-world performances, the ITS was evaluated in a controlled experiment at two geographically separated Air Force F-15 flying wings (at Langley AFB and Nellis AFB). A total of 26 subjects were selected from master and novice technicians. Two measures of troubleshooting proficiency were used in both the pre-test and post-test: a non-interactive written test (NIT) composed of mini (focused) troubleshooting scenarios, and verbal troubleshooting tests (VTT) which employed a type of structured, thinking aloud verbal protocol. In addition, technicians completed a post-tutor evaluation questionnaire. The technicians were tested for the standard I&C test station and, on post-tests, for Frankenstation—a test station none of the technicians had worked with before—to see how far technicians could generalize their troubleshooting skills.

Data showed that the tutored group outperformed the untutored group on both written and verbal post-tests, on both the I&C and Frankenstation test stations. These results are depicted in the graphs shown in Figure 10 on page 40. The effect of Sherlock II is defined as the difference between the control and tutored subjects on the five post-tests. The size of the difference is measured in standard deviations (SDs), with positive values indicating superior performance by the tutored group. Effect sizes of 0.3 to 0.5 SDs are usually taken as evidence of a successful education intervention. In this field evaluation, Sherlock II effects ranged from a gain of 0.6 SDs for the Frankenstation NIT to a gain of 2.78 SDs for the near transfer VTT. Twenty to thirty hours with Sherlock II yielded dramatic improvements in novice's ability to solve complex, novel troubleshooting problems.

The pattern of responses provided by technicians for their reaction to Sherlock II and their confidence about their troubleshooting skill suggests that technicians were generally pleased with Sherlock II.

Operating Environment

Sherlock II runs on 386 machines equipped with at least 16MB RAM, 20MB hard disk space, Windows 3.1, a Sony videodisc player, a Promotion video board and accompanying video driver software. It was developed using ParcPlace VisualWorks software and Object Technology, Inc.'s ENVY version control system.

Future System Plans

LRDC researchers are currently developing a tutor for F-15 flightline avionics. They are coordinating with BBN, who is developing a similar tutor for F-16 flightline avionics. Researchers also plan to evaluate the effectiveness of the reflective activities that have been implemented. Other future work is centered around extending the software framework to support peer collaboration.

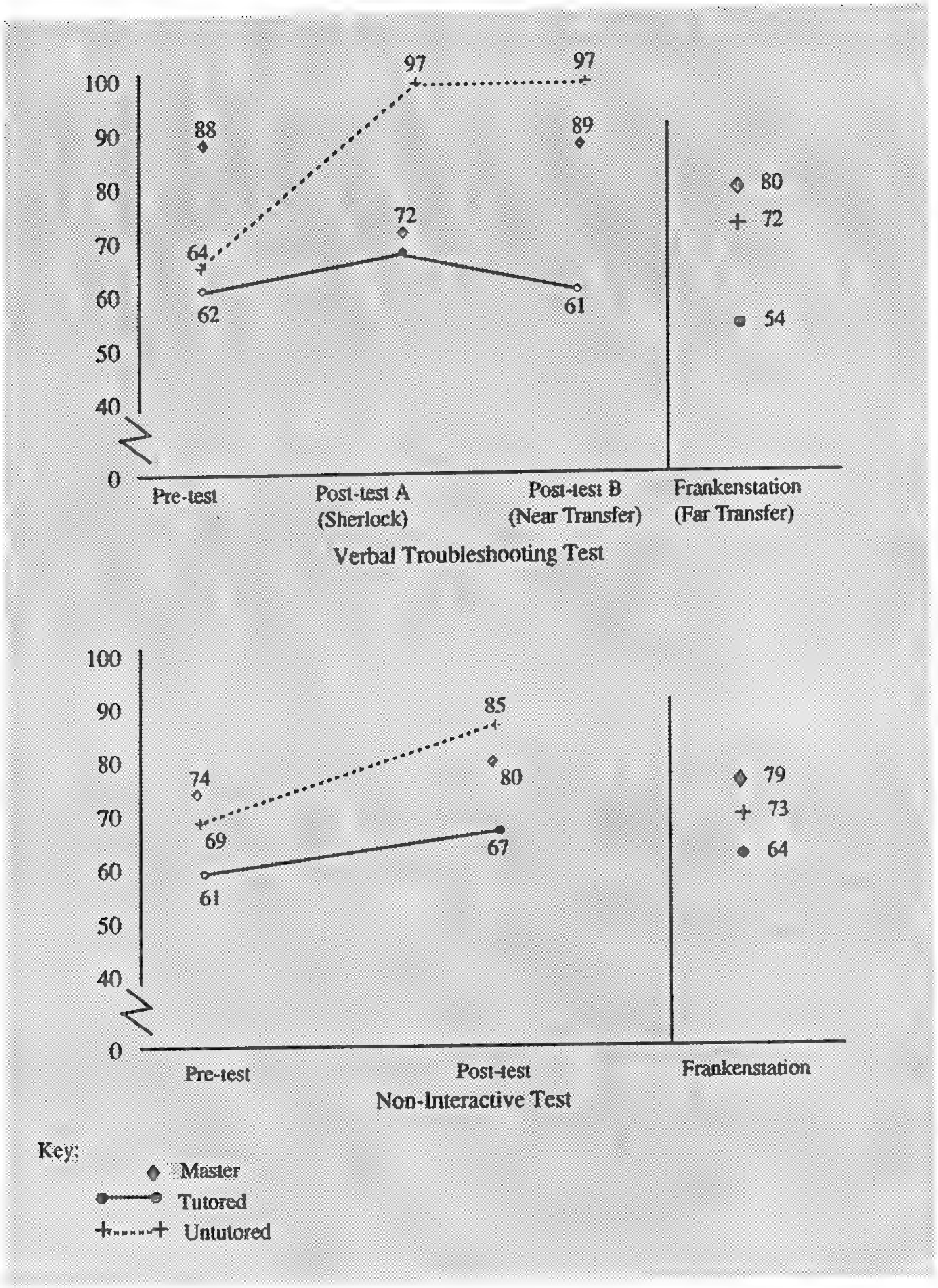


Figure 10. Results for Sherlock II Field Test

2.4.9 Intelligent Training Development Methodologies Project

The project provides the first merger of the principles of instructional design and development tools for use in virtual environment (VE)-based simulations. The objective is to identify the best principles for this. It supports exploring the value of instructional techniques that are unique to VE. The major product will be a prototype software toolkit for developing and delivering VR-based ITS.

Programmatic Background	This 6.2 effort has a funding amount of approximately $400K. In-house researchers are being supported through a contract awarded to Lockheed Corporation. Work began in May 1993 and is expected to complete in April 1995.
Planned Products	Software Development Toolkit. Prototype VR-based ITS for teaching orbital mechanics.
Approach	This information was not available.

2.5 Future ITS-Related Efforts

This section discusses two planned efforts: Improved Instructional Techniques for Intelligent Training Systems Project and the Virtual Interactive Technologies ITS Development System (VIVIDS) Project

2.5.1 Improved Instructional Techniques for Intelligent Training Systems Project

This project is necessary to support the upward transition of emerging knowledge about effective pedagogical strategies for automated instruction. Armstrong Laboratory and the AFOSR jointly sponsor a state-of-the-art research laboratory studying the pedagogy of automated instruction; that is, the Co-Lab discussed in Section 2.4.1 on page 7. The function of this project will be to implement prototype automated instructional systems in Air Force domains, based on instructional strategies developed in TRAIN, and other pedagogical research laboratories. These prototype instructional systems will include part-task trainers, ITS, digital video interactive (DVI), hypermedia, and multimedia environments. Once this research is completed, knowledge of how to use instructional strategies will enable the Air Force to train its personnel far more effectively and efficiently. The Air Force will be in a position to transition this technology into industry and the public sector, making training and education fundamentally more reliable and effective.

Programmatic Background	The Request for Proposal (RFP) for this 6.2 effort was released March 1, 1994. The estimated contract award date is July 1, 1994. The contract duration will be 48 months, with total funding estimated at $3M.
Planned Products	To be determined.
Approach	

The technical approach will be to create a review team consisting of AI specialists and Air Force personnel to review current literature for existing research and approaches, identify Air Force training needs, establish contact with major command points of contact, develop design documents for experimental instructional approaches, and develop and validate these instructional approaches. The experimental approaches will be monitored by AL/HR as funds are allocated to the experimental design and development phase of this project. Each interim hardware and software instructional delivery system prototype will be field tested, evaluated, and validated at one or more test sites. The prototype will also be demonstrated to interested parties (the training community of the military services and the civilian training sector). Major tasks include front-end analysis, development of design documents, development of interactive instructional software, and evaluation of experimental systems.

2.5.2 Virtual Interactive Technologies ITS Development System (VIVIDS) Project

Declining DOD resources increase the challenge in continuing to train US forces to the highest standards. More efficient ways must be found to rapidly develop and deliver training to US troops for such tasks as maintaining and operating complex weapon systems and quickly responding to hostilities in a growing variety of possible environments. Limitations of current training technologies result in training development and delivery systems that are very costly, time consuming, slow to respond to user needs, and produce training devices that quickly become obsolete. Related research into possible training applications for VEs at the Armstrong Laboratory has demonstrated this technology's capability for effectively training some of these cognitively complex tasks. Combining these findings with knowledge gained in other research conducted at the Armstrong Laboratory about how to quickly and efficiently build ITS using authoring shells provides the basic research necessary to specify this program. This effort will extend these findings and address the need for producing more efficient training development and delivery systems. The resulting payoff to the DOD will be a prototype ITS authoring shell incorporating VE technology. It will help training designers develop and deliver cost-effective automated instruction. Also, it will provide a taxonomy for applying VE-based instruction to DOD technical training, and specifications for an operational VE-based ITS authoring shell.

Programmatic Background A *Commerce Business Daily* (CBD) announcement was made in April 1994, requesting proposals from potential contractors by March 1995. The expected award date is August 1995. This 6.3 effort is allocated approximately $5M.

Prior Work This effort builds on work conducted in a prior project (called Advances In Virtual Reality). The objective of this project was to explore the instructional potential of VR as an interface for simulation-based training. It focused on two areas of R&D that support this architecture: (1) virtual world interfaces that greatly enhance the

man-machine interaction within ITS development with high fidelity 3D graphics, and (2) multi-functional knowledge bases that facilitate knowledge acquisition and implementation in the expert and instructional models of ITS.

The products of the work were a Systems Analysis Requirements Report that details the functional requirements for instructional development and delivery in a virtual environment. This report also provides an evaluation of the state of the art in hardware and software tools necessary to achieve the required functionality and how this will affect the final deliverable. A prototype virtual environment of an orbital mechanics domain was completed and demonstrated. A preliminary version of an training assistant component was also completed.

Experiments with console operations and spatial navigation in a VR yielded evidence that people can learn in a VR, and that knowledge and skills learned are transferred to the real world.

Planned Products Specifications, manuals, software products, prototype tutorials, and technical reports.

Approach An overview of the approach is given in Figure 11 on page 44. During the critical experiment phase of this project, researchers will first develop the initial specifications for a VE-based ITS authoring shell and write the alpha version of the software. Then, this version will be tested in the laboratory by building two prototype tutors with it. The software will be revised according to the feedback obtained in these tests. This will be the beta version of the ITS authoring shell. Then, as an advanced technology transition demonstration (ATTD), this version will be used to build approximately three prototype tutors at a variety of user testbed sites. Each of these tutors will be evaluated and the ITS authoring shell will be revised accordingly. As the prototype tutors are built, researchers will also develop a taxonomy and guide for applying the VE-based ITS authoring shell to technical training. The final step will be to transition the beta product, the guide, and specifications for an operational version of the ITS authoring shell to users and system program offices..

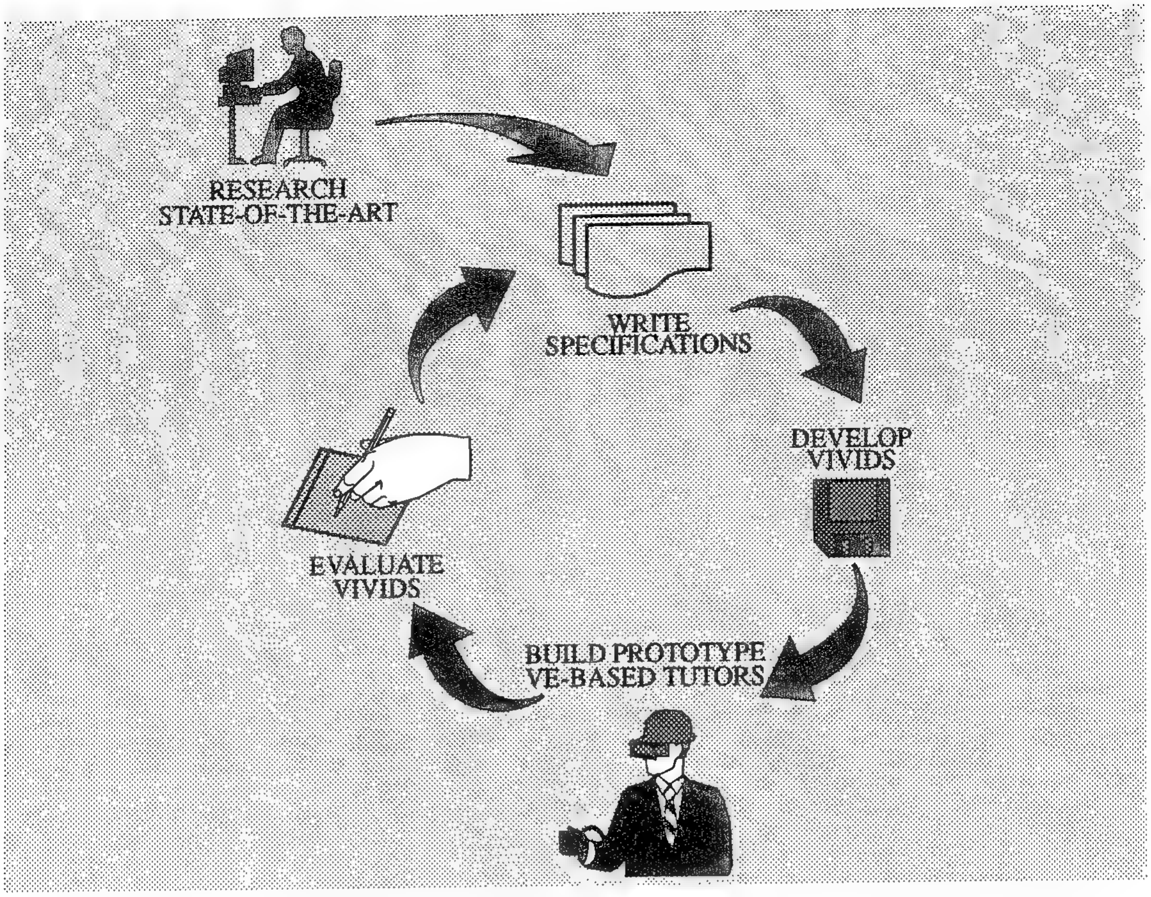


Figure 11. VIVIDS Project Approach

3. AIR FORCE HUMAN SYSTEMS CENTER

3.1 Mission and Role of the Human Systems Program Office

The Human Systems Center, San Antonio, Texas, is one of four Air Force Materiel Command Product Centers. The Human Systems Program Office provides system acquisition for the Center. Specifically, the mission of the Human Systems Program Office is to provide advanced performance, survival, and training technologies to the United States and allied air and ground crews through development, production, the sustainment of human-centered systems, and services including life support, aircraft escape, computer-based training, chemical defense, aeromedical, Air Force uniforms, mishap analysis, and environmental technology.

The Human Systems Program Office is productizing ITS technologies developed by Armstrong Laboratory in response to requirements from other commands. Currently, it is executing an ITS program for Air Combat Command (ACC) and finalizing the requirements of another program for Air Force Special Operations. It does not conduct R&D.

3.2 Summary of Past ITS-Related Work

The Human Systems Program Office has conducted several programs in the training area prior to the current ITS programs. However, no previous programs have been specifically executed for the development of intelligent tutors.

3.3 On-Going ITS-Related Tasks

Currently, ITS-related work by the Human Systems Program Office is following two directions. One direction is the development of production quality systems to meet previously identified ACC requirements. The other direction is a series of activities designed to support ITS users.

3.3.1 Maintenance Skills Tutor (MST) Project

The tutors developed in this project are the culmination of a number of efforts to transition technologies from Armstrong Laboratory into ITS production. The ACC began

to solidify its requirements in the late 1980s and a formal ITS acquisition program was initiated in 1990. To meet these requirements, the Human Systems Program Office is developing a series of ITS. These ITS will provide students with individualized troubleshooting training in a coached practice environment. In this way, the amount of personalized training available to each student will be increased, potential danger to personal or equipment eliminated, and the effect on senior maintenance personnel training time lessened. Most importantly, students will be given the opportunity to practice troubleshooting on problems that may occur infrequently in their jobs but that exercise critical troubleshooting skills.

MST is also active in process improvement for the development of streamlined tutor development methodologies.

Programmatic Background

The acquisition program started in 1990 and is currently scheduled to complete in 1999.

To execute this program, the Human Systems Program Office has assembled an Integrated Product Team composed of the scientists who did the original research, specialists in cognitive psychology and software engineering, personnel who cover the spectrum of acquisition professional disciplines, and several expert aircraft troubleshooting technicians from ACC. This government team will be supplemented with contractors that provide highly specialized skills in software design and development. The contracting organizations have yet to be determined. The point of contract is Dr. Jeffrey Kantor at the Human Systems Program Office.

Planned Products

Plans are to field the following ITS for developing troubleshooting skills:

- Early 1994, the F-15 Avionics Off-Equipment Manual Test Stand ITS.
- Late 1994, the F-15 Pneudraulics ITS and an F-16 Avionics C Shop ITS.
- In 1995, the F-15 Avionics A Shop ITS and C Shop ITS.
- In 1996, the F-16 Avionics A Shop ITS.
- In 1997, the F-16 B Shop ITS.
- In 1998, the F-16 Engine ITS and F-16 Pneudraulics ITS.
- In 1999, the F-15 Environmental/Electrical Systems ITS and the F-16 Environmental/Electrical Supply ITS.

These systems all will be based on the Sherlock ITS. A description of Sherlock, its architecture, and evaluation can be found in Section 2.4.8

Approach

The MST program will field 11 different versions of ITS. Each version will train troubleshooting skills for a specific operational system. Although every attempt is being made to capitalize on the opportunities to reuse ITS components from one training domain to the next, to a large degree, each ITS development is a separate project.

The basic ITS development approach starts with the following steps:

Step 1: *Data Collection.* The first step in developing an ITS is the collection of basic data. These data will be used to develop all facets of the tutor—from expert model and coaching messages to the student interface. The Procedure, Action, Result, Interpretation (PARI) technique developed by Armstrong Laboratory is used for the data collection. This collection process is composed of nine stages, as shown in Figure 12 on page 47. In general, the first four stages determine the group of experts that will further contribute to the task analysis and identify at a general level the problem-solving tasks and associated cognitive skills that are to be considered as instructional targets. These tasks and skills serve as the initial focus in the development of specific problems to be solved during the PARI interviews. The final five stages involve the basic expert, intermediate, and novice interview sessions, the follow-up rehashes, and review and analysis of the data. The overall technique is based on the Subject Matter Expert (SME)-problem pair.

Stage 1 - Subject Matter Expert (SME) Selection & Orientation of Researchers. On-site SMEs are identified by the research team and asked to describe their job specific training and experience. SMEs then orient researchers to the job specific equipment systems, relevant TOs, site-specific maintenance practices, and problems they have encountered while training technicians of different skill levels.

Stage 2 - Determine Focus of Training. SMEs determine the training foci for their job area by listing specific maintenance tasks and equipment systems that they consider to be "cognitively complex" as well as provide specific examples of equipment malfunctions.

Stage 3 - Generate and Consolidate Problem Types. SMEs independently generate exhaustive lists of potential faults using the foci from Stage 2 and collectively consolidate the faults into categories based on similar knowledge and skills required for problem solution.

Stage 4 - Specific Problem Design. SMEs are assigned to problem categories developed in Stage 3 and generate representative problem descriptions and statements individually. The problem description describes the spdsecific location of the fault in the system, symptoms related to the fault, and alternate fault locations (i.e., locations in which a fault may be placed and continue to manifest the same symptoms). The problem statement describes the symptoms of the fault as they would be presented to the technician on the job. Researchers describe and discuss with SMEs:

a. Criteria for good problems.
b. Example problem descriptions and solution paths.
c. Upcoming PARI sessions.
d. PARI rehashes.

Stage 5 - Development of Problem Generator's Solution Path and Alternate Paths. Each SME works one-on-one with a researcher to document, in PARI format, his preferred solution path to each problem that he has generated, step-specific block diagrams, and alternate PARI steps. SMEs are attach copies of relevant equipment panel diagrams from TOs to simulate on-equipment fault isolation.

Stage 6 - Generation of Expert Solutions. SME-researcher pairs interview other domain experts to generate PARI solutionis (including alternate PARI steps) for experts who are "blind" to the source of the fault. The roles of SMEs and researchers during the PARI documentation process are reviewed.

Stage 7 - Problem Set Review by Expert Problem Solvers. Selected SMEs review the set of problems developed at Stages 4 and 5 for completeness and accuracy. SMEs and non-interviewed experts rate the criticality of knowledge, skills, and strategies as well as rank order the problems in terms of difficulty and the utility of necessary cognitive skills.

Stage 8 - Generation of Novice and Intermediate Solutions. SME-researcher pairs interview novice and intermediate level technicians in order to generate PARI solutions by technicians of different skill levels.

Stage 9 - Final Problem Set Review. The final problem set is reviewed by senior experts (third-line supervisors and other advanced experts) with a broader experience base than the SMEs for accuracy, representativeness, completeness, and utility in training.

Figure 12. PARI Data Collection Stages

Step 2: *Data Analysis*. Once the raw data has been collected, the analysis phase begins. The primary focus of the analysis phase is to transform the raw data into tutor content. (The system elements that constitute "tutor content" are equipment simulation, coaching messages, expert system, assessment criteria, and problem selection criteria.) The analysis procedures that underlie the transformation of raw data into tutor content are inherently complex and are still in the development/refinement stages. Currently, the overall analysis process is decomposed into a series of sub-analyses. Figure 13 on page 49 identifies the data analysis stages.

Step 3: *Developing System Specification*. After the data have been analyzed, taking these data and developing requirements that will be used in the ITS software development ensures that the requirements of the users will be met. This methodology requires taking generic hardware and software requirements used within each ITS development and combining the unique domain specific requirements to develop a system specification that will be used throughout the ITS software development.

To support software modularity and reuse, the software approach requires a common ITS architecture. The ITS architecture breaks out functions into specific software areas that are identified as "Models." There are four different models that are used to define these functions. These models are the Coaching Model, Student Model, Expert Model, and the Curriculum Model. These models contain their specific functions and provide their information to other models based on the users' inputs. The development of an ITS architecture encapsulates each of these tutor functions, such as the simulation of the domain area, expert information on how to troubleshoot, assessment of a student during execution of a problem set, and other unique tutor functions that will provide the developers and maintainers the opportunity to quickly and easily make changes and modifications. Reusing tutor software from one system to another requires that the "generic" portions of the software models are independent of specific domain data. The use of fourth generation languages (4GLs) provides the tools necessary to integrate simulation, AI, and multimedia technologies in a harmonious application.

Project Status At present, there are on-going development efforts for six of the ITS that will be fielded. Development efforts range from clean-up for the first ITS to be fielded (the F-15 Avionics Off-Equipment Manual Test Stand ITS) to early data collection for the F-15 Avionics A Shop ITS. In addition to the actual ITS, a series of tools are being developed which will increase the efficiency of later efforts. These tools will improve various components of ITS development, including the cognitive task data analysis and software coding.

3.3.2 ITS User Support

The Human Systems Program Office began a series of annual user conferences starting in March 1994 at Brooks AFB, San Antonio, Texas. The purpose of this initial conference was to review ongoing and planned ITS development efforts, explore additional applications, and formulate cooperative efforts to satisfy multiple training needs. It provid-

Analysis I: Task Selection. To determine which jobs (and tasks within a job) are of sufficient cognitive complexity to warrant supporting an ITS.

Analysis II: Troubleshooting Activities Beyond Equipment Interaction. To identify any activities directly related to troubleshooting other than those that involve interacting with the equipment; to define a schema for their implementation into the tutor.

Analysis III: Equipment Troubleshooting Actions. To identify and categorize those troubleshooting actions that must be supported by the tutor that involve interacting with the equipment or with the technical data; to describe the manner in which those actions should be executed by the student and by which the resulting response should be generated.

Analysis IV: Equipment Definition. To identify and graphically represent the system components and identify the support equipment that is required for the simulation module of the tutor; to establish a class hierarchy of system components.

Analysis V: Equipment Characteristics and Behaviors. To identify each system component's characteristics (ID, location, system(s), (sub)systems, and behaviors (internal, external)) when the component is in either a faulted or unfaulted state; to provide a graphic representation of each component which includes any subcomponents that are on the active path and any labels, conditions, or values relevant to its behaviors; to identify which troubleshooting actions may be taken on which system components; to refine the class hierarchy so that classes reflect components with common behaviors and that subclasses are defined, when necessary, to distinguish behaviors that are different among class members.

Analysis VI: Equipment Displays. To identify all the equipment displays and the corresponding "hot spots" that are required to support both troubleshooting actions and coaching on the simulated equipment.

Analysis VII: Information Accessibility. To list the type of information sources that must be available to the student and to assemble copies of or create the actual information that will be available either on-screen or in hard copy.

Analysis VIII and IX: Global and Local Goal Structures. To define the production rules for the expert system that evaluates student actions and updates what is (or should be) known about the status of subsystems, functional areas and/or system components.

Analysis X and XI: How It Works Coaching Messages. To establish both the format and content of the system knowledge to be incorporated into the coaching function of the tutor, including descriptions of:

- The overall functions of a problem's active path and sub-paths.
- The general purpose of each system component.
- The operation, inputs, and outputs of each system component.
- The specific role of each system component in a problem.

Analysis XII and XIII: Assessment Scoring. To establish categories of troubleshooting violations, the point deductions associated with each category, and the set of actions within each problem that represent actual instances of those violations, e.g., the category of unsafe measurements may carry a deduction of 30 points from the Procedural Knowledge score and may be defined as some set of ohms measurements in one or more problems.

Analysis XIV: Problem Set Categorization. To establish the criteria for categorizing problems as "novice", "intermediate", or "expert" and, within those categories, for rank ordering the problems; to categorize and rank the set of core problems according to those criteria, to determine the number of additional problems that must be created in each category, to create the additional problems from the core set by varying the fault location and/or amount of information revealed by the problem statement; to verify that the equipment behaviors defined in Analysis V adequately cover the new faults.

Analysis XV: How To Test Coaching Messages. To establish both the format and content of the strategic and procedural knowledge to be incorporated into the coaching function of the tutor which includes descriptions of:

- System components tested since problem inception, as well as those eliminated or remaining to be tested.
- Completed tests and the interpretations of their results.
- Strategies for selecting components to test and how to test them.

Analysis XVI: How To Trace Coaching Messages. To establish both the format and content of the strategic and procedural knowledge to be incorporated into the coaching function of the tutor which includes descriptions of:

- Strategies for accessing the tech data needed to troubleshoot problems.
- Explanations on how to interpret the information in the tech data.

Figure 13. Purpose of ACC Data Analysis Activities

ed trainers, training managers, and planners with a chance to explore the variety of options available, and to meet others with similar training problems.

Attendees were asked to complete a survey questionnaire that would allow the conference staff to compile a summary of some of the applications in which ITS are currently being used and to facilitate shareware opportunities. The information was also used to facilitate a discussion of how ITS could be used to meet attendees training requirements.

4. ARMY MATERIEL COMMAND, MISSILE COMMAND

4.1 Mission and Role of Research Development and Engineering Center (RDEC), Guidance and Control Directorate (G&CD)

In the Army Materiel Command (AMC), Missile Command (MICOM), ITS work is being conducted by the Research Development and Engineering Center (RDEC), Guidance and Control Directorate (G&CD). At the highest level, the mission of RDEC is to plan, manage, and conduct research, exploratory, and advanced development for assigned materiel. This includes planning, establishing, and managing the MICOM programs to provide lifecycle systems engineering and production engineering, and executing management of computer resources embedded in battlefield automated systems. Additionally, RDEC serves as the AMC Lead Laboratory for Guidance and Control-Terminal Homing and for High Energy Lasers.

The mission of G&CD is to conduct research, exploratory and advanced developments, technology demonstrations, technology insertions, advanced technology transfer demonstrations, preplanned product improvements, and risk reduction programs in several guidance and control technology areas.

4.2 Summary of Past ITS-Related Work[1]

The ITS development effort at the MICOM was initiated in 1984 in response to a specific need identified by the HAWK Air Defense Weapon System Project Office. Specifically, the need existed for a device that could be embedded in the weapon system and used to develop and maintain critical skill levels required to operate the weapon system in a battlefield environment. Results from this initial R&D effort provided a theory of operation and functional design of the Intelligent Embedded Operator Assistant (IEOA).[2]

[1] This information was supplied by Dr. Willard M. Holmes in a personal communication entitled "An Overview of Intelligent Tutoring System Research and Development at the U.S. Army Missile Command, Redstone Arsenal," received April 28, 1994, by the author.

[2] IEOA is an ITS whose purpose is to support operator skill maintenance in an embedded surface-to-air missile system (HAWK) simulator. Its development was supported by the RDEC and the first version has been operational since 1987. Under 1991-1993 NASA SBIR sponsorship, IEOA was moved from a LISP-based expert system environment to a Unix, C++ environment running on a Silicon Graphics workstation.

The scientific and technical contribution made through the IEOA development effort was in two areas of technology: the development and use of a student model to individualize tutoring operation, and second, the use of embedded scenario generation to adaptively synthesize a task or scenario as required to develop and maintain the critical skills of the weapon system operator.

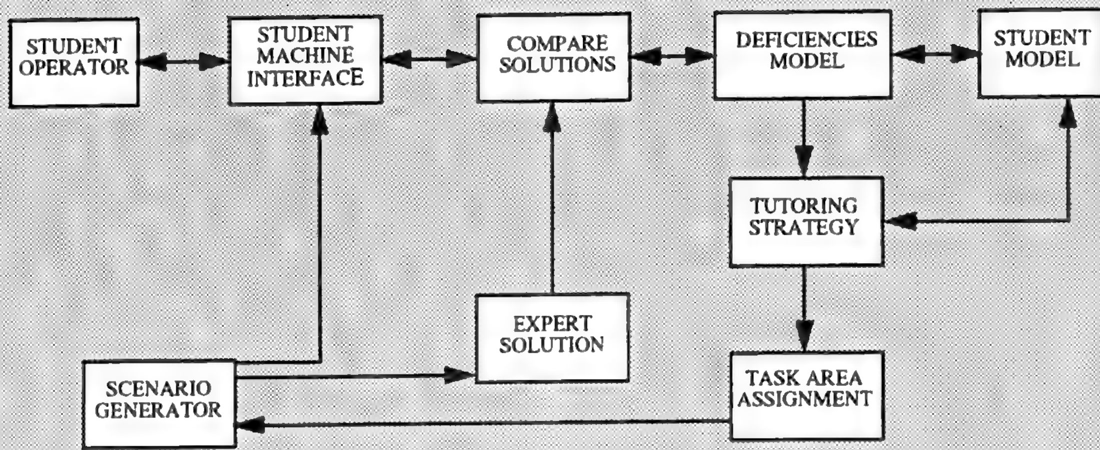
Prior to the IEOA development effort, ITS R&D was confined to the laboratory environment and primarily focused on validating hypotheses in learning theory. The major use of computer-based training or computer-based instruction was in the open-loop mode or “canned program” approach. This mode of operation means that the subject matter material is presented without regards to the individual’s skill or knowledge deficiency in the subject area, and without interaction by the student during program operation. Furthermore, the teaching methods used in presenting the instructional material to the student are not individualized to optimize skill development efforts for the particular student. Without the use of a student model, effective and efficient use of individualized instruction and teaching methods is not feasible.

The use of a scenario component generator in an embedded tutoring system permits the synthesis of a task individualized to the training needs of the student. The scenario generation process makes use of knowledge about the human student that is contained in the student model. The knowledge in the student model evolves during the process of tutoring and assessing the system operator’s performance. A knowledge base relative to the individual student’s deficiencies in skill and concept development is maintained in the student model. Expert rules are used to analyze the conceptual deficiencies of the student as indicated in the student model. Results from the performance analysis are used to generate a scenario that address the particular skill or concept deficiencies of the student. The scenario generation is accomplished on line and in real time.

An overview of the IEOA ITS is given in Figure 14 on page 53.

4.3 Overall ITS R&D Program

In 1989, the Army Technology Base Master Plan [AMC 1991] was issued to provide guidance to the Army Laboratories and Research Centers to focus the technology base on the most critical war fighting needs; it designates AI as one of 13 key emerging technologies. Subsequently, in late 1990, the AMC’s Deputy Chief of Staff for Technology Planning and Management directed the preparation of a comprehensive master plan for AI which would serve as a framework from which the AMC could manage and execute AI



Functionally, the IEOA operating process can be viewed as starting with the *Scenario Generator*. The *Scenario Generator* generates a task or scenario that is presented to the *Student Operator* through the *Student Machine-Interface*. Concurrently, the scenario is presented to the expert in the *Expert Solution* module. The choices and related actions of the student in response of the scenario are transmitted to the *Compare Solution* module via the interface. The appropriate action taken by the expert is compared with the student action in the *Compare Solution* module. The differences between the appropriate action generated by the expert and the student action is generated in the *Compare Solution* module and transmitted to the *Deficiencies Module*.

A model of the student's deficiencies associated with the present scenario operation is generated in the *Deficiencies Model* module. Elements of the *Deficiencies Model* are transmitted to the *Student Model* for developing particular knowledge about the individual student and a history of the student's performance. In addition, information generated the *Deficiencies Model* is transmitted to the *Tutoring Strategy*. With information about deficiencies in performance on the present scenario and background and history of the student's performance from the *Student Model*, an appropriate tutoring strategy for the student is generated by the *Tutoring Strategy* module.

An appropriate tutoring strategy includes specified problems of tasks for the student to accomplish to improve performance or correct identified deficiencies. The *Task Area Assignment* module receives information from the Tutoring Strategy module and identifies the specifics of a required task. The specifics of the required task are transmitted to the *Scenario Generator*. The new scenario is transmitted to the student and the *Expert Module*. The process is continued until the skills of the student are developed to a predetermined level or the expert solution level.

Figure 14. IEOA ITS Architecture

technology development into the next century. At its lower level of detail, the AMC AI Master Plan identifies 13 technology subareas as critical to Army needs. One of these technology subareas is that of ITS, and MICOM was specified as the lead player.

Based on the results obtained from demonstration systems, effective ITS in limited domains can be constructed and progress in developing such systems requires advancements and building on past accomplishments in both theory and technology in broad fronts. Accordingly, with respect to ITS, the goal of the AMC AI Master Plan is given as being: "To advance and apply the state of the art in ITS to meet the training needs of the Army into the 21st Century." The three objectives that support this goal are to (1) advance the state of the art in ITS, (2) apply the state of the art in ITS, and (3) identify and implement major thrusts to further facilitate advancing and applying the state of the art in ITS. A summary of the functional requirements identified within the established objectives is given in [Holm 1991] and reproduced in Table 1 on page 55. The Master Plan continues by describing the major technology thrust program for ITS as a development effort that will demonstrate the confluence of AI technologies in advancing ITS development. This thrust program includes three elements:

- An ITS architecture development effort for implementing and integrating the top-level ITS modules to meet Army training needs,
- A shell for use in a development environment which includes developing and implementing ITS functions, and
- A micro-chip based ITS for accomplishing symbolic and numeric processing in a delivery environment.

The AMC AI Master Plan was never published, but has influenced ITS-related work of both MICOM and the Army Simulation, Training, and Instrumentation Command (STRICOM).

4.4 On-Going ITS-Related Tasks

The G&CD currently is sponsoring a single SBIR effort. Another SBIR effort was completed in the course of preparing this report. Both efforts are discussed below.

4.4.1 ARPA Intelligent Tutoring System Project

The goals for this Phase II project, which have guided the work in Phase I and will continue to guide the system throughout its development, are as follows:

- To develop a stand-alone training device embedded within operational air defense weapon systems.
- To provide modular independence so that one module of the ITS can be altered or replaced without affecting the others.
- To develop a generic system, reusable for many different scenarios and applicable to other domains.

More specifically, Phase II is concerned with taking the architecture defined in Phase I and transitioning to detailed design and prototype implementation. The application domain is that of the Avenger Table Top Trainer.

Table 1. AMC AI Master Plan ITS R&D Activities

	Advancing State-of-the-Art Areas	Applying State-of-the-Art Areas
Near Term (1-2 years)	Generic ITS architecture for individual and crew training Tools for real-time ITS implementation Modeling for temporal reasoning and reasoning under certainty Cognitive modeling for teaching, learning and causal reasoning	Analyze Army training requirements Prototype development of ITS with generic architecture for individual training Verification of validation of ITS operations
Mid Term (3-7 years)	Shell architecture development for real-time ITS applications Modular generic architecture for force-level training systems Adaptive human-machine interface using student-based sensor data Cognitive model development for knowledge and skill transfer	Develop ITS technology demonstration for individualized operator training system Develop architecture for ITS operating as an embedded training device
Long Term (8-17 years)	Micro-chip technology for ITS delivery platform Theory and architecture development for real-time ITS with multiple and distributed knowledge bases Cognitive modeling for diagnostic evaluation of student's mental model, misconceptions, and reasoning	Distributed ITS for team training with crew members geographically displaced Develop portable, personalized ITS adaptable to specialized domains with modular components for domain knowledge

The major innovation in the work lies in diagnosing student deficiencies in the domain using a pioneered fault isolation technique. A student deviation from the expert solution is detected when that specific solution element is activated and no comparable student action takes places. That solution element has a one-to-one mapping to the Domain Network and the Student Model, as illustrated in Figure 15 on page 56. The unused node,

together with any child nodes, forms the Suspect Ambiguity Group (SAG). Knowledge about previous student performance is then used to reduce the number of suspect nodes. A number of sub-scenarios can then be generated to test specific nodes. This testing sequence is guided by factors such as the cost of testing. This fault isolation approach is intended to improve the overall effectiveness and efficiency of the tutoring.

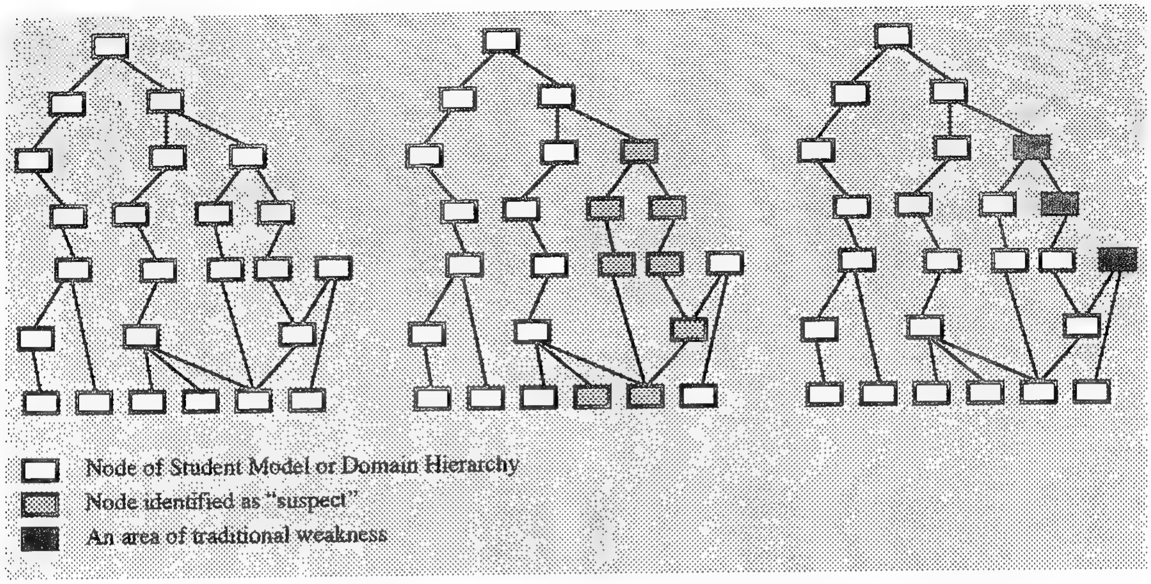


Figure 15. Initial Diagnosis Steps for the Fault-Isolation Technique

Programmatic Background

This work is being sponsored by the U.S. Army Missile Command under contract #DAAH01-94-C-R008. It is funded through an ARPA SBIR, ARPA Order No. 6685. The contract was awarded to Global Information Systems Technology, Inc. Begun October 1, 1993, Phase II is due to be completed September 30, 1994. The funding amount for Phase II is $388,885.

MICOM personnel will support the contractor is several ways, most notably in selecting a testbed system for training diagnosis and maintenance concepts, and in conducting evaluations and demonstrations of the prototype system. Additionally, Global researchers will be supported by consultants from the University of Pittsburgh. The Principal Investigator is Dr. John Maguire from Global.

Planned Products

Prototype ARPA ITS for the Avenger Table Top Trainer domain.

Prior Phase I Work

The work in Phase I focused on two areas of development. The first was preparation of an ARPA Intelligent Tutoring System (AITS) Architecture Design Document. This document details a proposed approach to implementing an ITS for a family of weapon systems, based on both the AMC ITS Master Plan and on the IEOA (see Section 4.2 on page 51) begun by the Army in 1988.

The second area of development was a demonstration system. It simulates an operator scope and provides intelligent guidance to the student maintain-

ing skills as a tactical operator of a weapon system. The demonstration program models rudimentary functionality of the tutoring system.

Approach The technical approach to Phase II includes seven main tasks:

Task 1: *Review and Expand the ARPA ITS Architecture.* As a prerequisite to developing each of the functional modules, the ARPA ITS architecture developed during Phase I will be reviewed. This architecture will provide the framework for implementing Phase II software modules. **This task has been completed.**

Task 2: *Revise and Publish the ARPA ITS Architecture Report.* This task requires including the new architectural changes, made under Task 1, into the original Phase I report. Concurrently, as part of Task 3, consultants from key universities will review the architecture and suggest possible changes. The architecture report is expected to be a living document that reflects the growing understanding of this technology and new methods to achieve a truly generic and reusable system. **This task has been completed.**

Task 3: *Investigate Cognitive Science Issues and Ramifications to ARPA ITS Design.* The intention of this task is to bring the two fields of Cognitive Science and Computer Science into a single ITS. Consultants who are experts in the ITS design and development and have strong backgrounds in cognitive science will evaluate the system architecture and implemented software modules, and make an assessment of the effectiveness of the delivered training for the Avenger Missile System. **Work on this task is continuing.**

Task 4: *Design and Implement all ARPA ITS Software Modules.* The fourth task is to implement the software modules of the ITS design. Software will be written in C++ to facilitate portability, particular attention will be paid to ensuring a truly modular system that supports code-replaceable modules. **Work on this task is continuing.**

Task 5: *Acquire Select Domain Knowledge and Author Instructional Test Database.* This task addresses knowledge acquisition and authoring functions for the selected domain. It will support demonstration of the final software in a real-world domain. The collected knowledge will be encoded into a domain hierarchy and expert solution.

Task 6: *Verify System Performance.* Various aspects of the work will be evaluated: the task analysis, proper functioning of the software modules, and the resulting ITS delivery to the student (effectiveness and efficiency in training). To this end, a series of one-on-one formative evaluations of the product will be conducted. "Gross" problems to the courseware will be corrected immediately. The results of this evaluation will be compared across of range of representative students. Unobtrusive one-on-one evaluations to assess the validity of these results in a more natural setting will be performed. Several rounds of obtrusive and non-obtrusive evaluations will be performed.

Task 7: *Demonstrate the ARPA ITS Software.* The final objective is to demonstrate and deliver to the Government the ITS using the test domain database. MICOM personnel will have been provided with the ITS software and database during the

course of this project, with software and database updates provided as appropriate. Not only is this approach designed to encourage greater Army participation in the development process, it will minimize any risks involved with transporting the system to the demonstration site.

Potential Phase III Follow-On

Phase III will involve additional design work and implementation of baseline system functionality. Further research is required in certain areas before finalizing the design:

- Cognitive approaches. Working with consultants from the University of Pittsburgh, Global will enhance functioning of the Tutoring Strategy Module and examine the proposed design for Compare Solutions (see the ARPA Intelligent Tutoring Systems (AITS) architecture description in the next section).
- Student-expert deviations. A generic definition of a significant difference in behavior between the expert and the student will be determined.
- Modularity. Software engineers will minimize coupling of modules with well-defined inter-module message and object-oriented programming techniques.
- Intelligent interface design. New research in intelligent interface design will be reviewed for enhancement to the AITS interface.

4.4.2 ARPA Intelligent Tutoring System (AITS)

The ARPA ITS (AITS) automates the role of an instructor who evaluates the performance of a student in a simulator environment. The application domain is that of the Avenger Table Top Trainer. Conceptually, the instructor performs the following three steps:

- Observe the student's actions during the simulation and spot student mistakes.
- Determine why the student made the mistakes.
- Tutor the student to correct these mistakes.

In similar fashion, the AITS duplicates these functions with its three modes of operation: (1) Observation, (2) Diagnosis, and (3) Tutoring. The Observation mode presents a complete scenario to the student. Scenario selection is based on the student's previous performance (if any) and driven by the goal of maintaining student skills. Ideally, the student interacts with the simulation without any interference from the system. AITS makes no explicit judgement of the student; however, the AITS is always collecting and analyzing data from the simulation. It determines when the student commits errors and the severity or criticality of those errors. The scenario can be stopped or displayed in accelerated fashion if the student makes serious operational errors. A scenario may be concluded if, for example, any of the following occur:

- A certain number of opportunities for action have been executed without student response.
- A certain amount of time has expired.
- The student has failed to perform critical actions or performed inappropriate ones.
- The student has successfully performed all actions that the expert would (that is, the student performs similarly to the expert solution).

In the Diagnosis mode, a shorter, simpler sub-scenario is run to help isolate the cause of the student's mistake. Completion of the scenario occurs upon satisfaction of ending conditions specified initially or when another student deviation from the expert solution is detected. Here again, a deviation initiates the fault isolation process to determine its cause. A sub-scenario is also run in the Tutoring mode, but here interaction with the student may not be necessary and completion of the scenario occurs upon satisfaction of ending condition or if student responds incorrectly (as judged by the Tutoring Strategy module).

Architecture Figure 16 on page 59 shows the proposed ITS architecture developed during the Phase I effort. This figure also indicates the modules that were developed for the demonstration system.

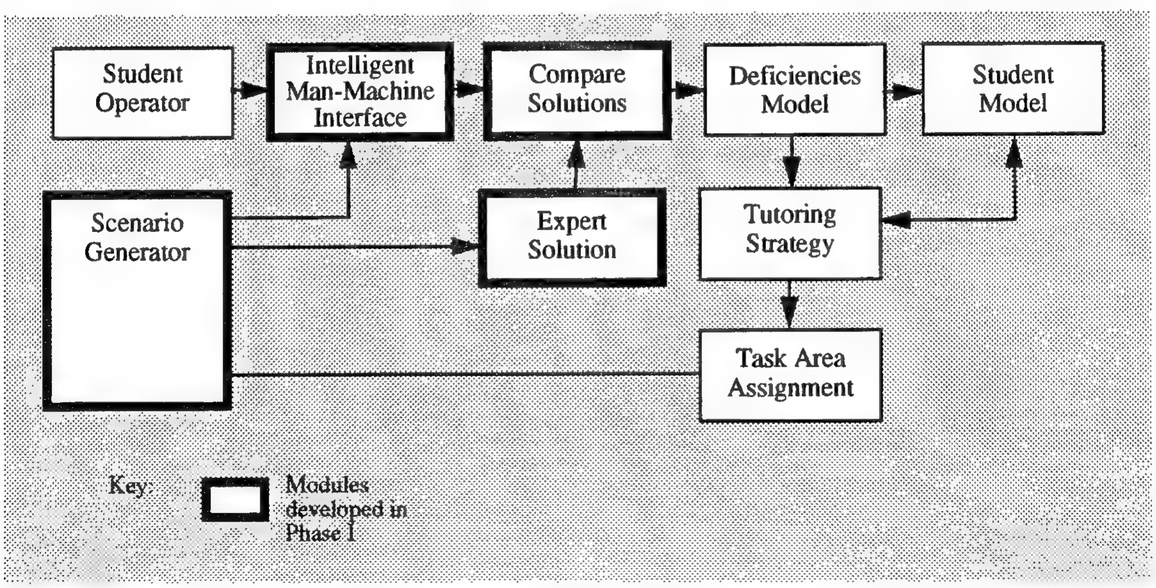


Figure 16. Proposed ARPA ITS Architecture

Scenario Generator This module performs whatever calculations and decisions are necessary to deliver a selected scenario. (The specific scenario, its characteristics, and initial conditions are provided by the Task Area Assignment module.) The student interacts directly

with the scenario. The interactions consist of the student observing the display and ancillary outputs (that is, synthesized sounds and digital voices) and responding appropriately (specifically, by toggling switches, entering data at the keyboard, or selecting items via a mouse, etc.) As the Scenario Generator creates a set of events (for example, foreground objects or events moving within the simulation space), the student may respond by taking various actions. Working in parallel, the Expert Solution module reacts by generating a set of its own actions. The streams of student and expert actions are sent to the Compare Solutions module. If a significant deviation exists between the two, the scenario is halted or, in some cases, runs to its logical conclusion, perhaps in accelerated mode.

Expert Solution This module receives a series of time-coded event messages from the Scenario Generator, generates a response to each scenario event, and transmits this to the Compare Solutions module. One of two approaches to determining the response will be taken. The first approach is the traditional one of querying subject matter experts on their troubleshooting strategy and encoding it as a series of production rules. The alternative approach is based on the Dependency Model. In this case, the subject matter expert will enter a series of connectivity and dependency relationships between system components, and other parameters that help in performing fault isolation. The Dependency Model can then be used in conjunction with a "diagnostic strategizer" to guide a technician in performing fault isolation and repair. Rather than select just one of these alternative, researchers may decide to provide both solutions (perhaps the rule-based one for novice students and the dependency model-based approach for advanced students).

Compare Solutions This module has two functions: (1) detect when a student has performed an action which deviates significantly from the expert solution, and (2) help reduce the search space of the Deficiencies Model in which the student error is eventually isolated. The module receives streams of input from the Intelligent Man-Machine Interface and Expert modules, these are forwarded onto the Deficiencies Model module, with a diagnostic message if the student has made a mistake.

Student Model The Student Model is a collection of data elements and processes which represent the best understanding the ITS has of the student's knowledge. It contains a history of his performance overlaid on a representation of the domain. It also contains data elements related to his ability and style of learning. The primary function of the module, in Observation mode, is to select which full scenarios to run. In general, this selection can be based on statistical or individual student data. The module supports Diagnosis and Tutoring modes by providing information on historical performance and, potentially, domain information to the Deficiencies Model module.

Deficiencies Model This module begins operation when the Compare Solutions modules has detected a student mistake, or when the student has successfully completed the scenario. In a sense, this module diagnoses either a student's mistake or the student himself, and sends appropriate instructions to the Tutoring Strategy module. In the first case, the module attempts to diagnose the cause of the student's mistake. To do this, it makes use of the Student Model which contains information on the instruc-

tional domain and the student's historical performance. Diagnosis may require that the system test the student further to isolate the student error and the module requests the Tutoring Strategy to run a sub-scenario for testing rather than tutoring purposes. It provides a description of the domain area that needs to be tested. When and if the module has isolated the student's error, it then informs the Tutoring Strategy module of that error. If an error could not be isolated, then all aspects relating to the mistake need to be tutored.

Tutoring Strategy

On receipt of messages from the Deficiencies Model module to run a sub-scenario, this module simply passes the request onto the Task Area Assignment module. If the Deficiencies Model module does isolate an error, it informs the tutoring strategy of the error, so that this module can develop a plan for tutoring the student appropriately. When the tutoring is finished, the history of the tutoring session is stored in the Student Model. The original scenario is rerun to see if the tutoring had the desired effect. This module then requests the Task Area Assignment module to generate parameters for the scenario.

Intelligent Man-Machine Interface

This module is responsible for processing student operator inputs, receiving Scenario Generator requests to display, and presenting the scenario displays. Student responses are time-stamped and converted into specific student actions, and then sent to Compare Solutions module.

*Task Area Assignment*This module receives information about which domain area to develop a scenario. It then assembles the proper set of initial conditions for the Scenario Generator and forwards these to the generation module.

Operating Environment

Silicon Graphics, Indy workstation running Silicon Graphic's IRIS 5.1 and X11/R5 Window System, Display Postscript, and Motif-based window manager. The ITS application software is written in ANSI C and C++, using IRIS Graphics Library.

System Development Plans

These include the following:

Instructional Database Authoring Interface

Current manual development of the instructional database necessitates the expertise of AI system programmers to create the domain hierarchy and associated data structures required. To gain user acceptance, an authoring system will need to be developed to allow instructional specialists to create these databases themselves without programming knowledge. Using approaches similar to those employed with TIE (a system designed to allow instructors to generate computer-based training lessons), researchers will design and implement an authoring interface to the Phase II ITS.

Demonstrate Reusability of Architecture

To ensure that the ITS architecture and design is re-usable, researchers will measure the cost of using an entirely new domain. They will explore and develop methods to reduce the cost of developing a specific domain application by 50% of the Phase II cost.

Model Multiple Student Positions for Distributed Training Applications

The technology has potential for being embedded in the big brother system to the Avenger Table Top Trainer, namely the Integrated Conduct of Fire Trainer (ICOFT), which includes multiple student simulator stations all controlled and monitored from a single instructor station. Such an ITS would simplify the job of the instructor in evaluating student performance and providing a standardized tutoring strategy tailored to the student's particular domain weakness and learning style. The current ITS design supports single position training. For distributed or team training, researchers will study, design, and implement the software for integrating multiple student positions.

Port the ITS Software from C++ to Ada

To support the embedding of this ITS technology within weapon systems will require that the ITS software itself be written in Ada. The C++ software will be ported to Ada. The Authoring Interface software will remain in C++ because it resides in stand-alone workstations and is not part of the weapon system.

4.4.3 A Neural/Expert Based Client Server Architecture for MITE ITS Project

The objective of the SBIR effort is to investigate a MITE (multi-node, task-sharing, expert-instruction) ITS system. The MITE system is responsible for determining task allocation strategies for members of a team who must work together over a computer network to accomplish an overall objective. It must adapt to changing skill and performance levels of the team members and continually reallocate the tasks to ensure optimal team performance. To best allocate tasks among team members, the MITE system should maintain internal models of each member's capacity and knowledge relating to the tasks which that member may be allocated. The objective of this SBIR effort is to investigate a MITE ITS, whereby the system can also provide expert recommendations and instruction for team members to improve performance.

In this Phase I study, while investigating the application of hybrid neural network and knowledge-based strategies to the problem of MITE systems, researchers will also look for foundation technologies that can be applied to current or future commercial products with high potential returns. Designing the MITE system with an object-oriented client-server architecture provides the necessary reusability of code objects for a variety of application domains. For example, the complete MITE system can be used for both Army weapon systems, such as Avenger, and large-scale manufacturing applications. The task allocation objects can be used within MITE systems or for single user project scheduling. The neural network objects developed for the task allocation module can also be extracted and used for a variety of other optimization problems.

Programmatic Background

This effort is sponsored by AMC under contract number DAAH01-93-C-R327. The contract was awarded to Charles River Analytics, Inc. The period of performance is from August 1993 to February 1994. The contract is for a six-month fea-

sibility demonstration with expenditures not-to-exceed $50K. Dr. Jim Mazzu is the Principal Investigator.

Planned Products Proof-of-concept ITS. Requirements specification for a hybrid MITE ITS operating on a full three-node network of personal computers.

Approach The effort consists of four tasks:

Task 1: *Investigation of MITE ITS Architectures*. Researchers will investigate and identify object-oriented architectures for a MITE ITS which operates in ITS, full team and pseudo team (where the system simulates a team member) levels, with the hybrid integration of neural networks and expert system strategies to provide the required system intelligence. **This task has been completed. Researchers have identified the important classes and associated object properties needed for representation of team objectives, tasks to perform, and required resources.**

Task 2: *Development of Hybrid MITE ITS Modules*. Researchers will investigate the complementary use of neural network and expert systems to provide intelligence to the necessary MITE ITS modules for student modeling, expertise modeling, diagnostics, instruction, task scheduling, and human-computer interfaces. Appropriate neural network paradigms and architectures will be identified and developed, in addition to performing the necessary knowledge engineering to develop the required expert system rule bases. **This task has been completed. Researchers have shown how competitive neural network architectures will be applied to the problem of task scheduling, how those architecture must be expanded to provide optimal resource allocation, and demonstrated how the automatically generated schedule is interpreted by a fuzzy logic rule base.**

Task 3: *Identification of Task-Sharing Domains*. Researchers will identify three separate task-sharing domain problems which are applicable for the proposed hybrid MITE ITS. Charles River Analytics' current research in nuclear plant monitoring, operator assistants, aircrew task allocation, and the application's potential for commercial marketing will be considered. One of the identified task-sharing problem domains will be selected for the initial demonstration prototype. **This task has been completed. A system called PMS/Avenger has been investigated for potential communication and task scheduling coordination between remotely located Avenger teams which must work together to accomplish a common objective. Corresponding tasks and resources have been identified for application of the MITE system to the Avenger team.**

Task 4: *Development of a Prototype Hybrid MITE ITS*. Researchers will develop a prototype hybrid MITE ITS with single-node PC network operations for a selected task-sharing application. The executable version of the hybrid system development environment NueX/CLIPS will be delivered at the end of the contract along with the source code for the prototype MITE ITS. **This task has been completed. A prototype has been developed of a genetically evolving neural network paradigm by applying the genetic algorithm to gene strings which represent neuron connection paths, activation functions, and weight change functions.**

Task 5: *Requirements Specification and Documentation.* Researchers will define the requirements for a hybrid MITE ITS operating on a full three-node network of personal computers, in addition to documenting the Phase I results in a Final Report. **This task has been completed.**

Potential Phase II & III Follow-On

On the basis of the successful Phase I results, the researchers recommend the following:

- Development of a full-scale research prototype of the hybrid multi-node, interactive, task-sharing, expert-instruction (MITE) system. This involves expanding the object class diagrams, developing state transition scenarios, and implementing the object-oriented architecture in the hybrid development framework. The object-oriented intelligence framework provides fundamental building block objects for neural networks, genetic algorithms, fuzzy logic, and Open Sesame! learning capabilities.
- Refinement of the hybrid neural network and knowledge base architecture developed in Phase I in order to handle the faults which may occur during multi-node network operations. The researchers' experience with fault-tolerant systems will provide the basis for this effort.
- Incorporation of additional neural network paradigms in order to improve the modeling capabilities the system requires in order to properly reallocate resources after diagnosing performance of the resources on the tasks.
- In-depth application of the MITE system to the team of networked Avenger weapons, in addition to an extended development effort to apply the MITE task scheduling methods to general project management and resource allocation.

4.4.4 MITE ITS

Information about the MITE system is proprietary to Charles River Analytics, Inc.

5. ARMY SIMULATION, TRAINING, AND INSTRUMENTATION COMMAND

5.1 Mission and Role of STRICOM

The U.S. Army Simulation, Training and Instrumentation Command (STRICOM) is an AMC Major Subordinate Command. Activated in August 1992, the STRICOM role in the Army's vision of the future was outlined in a recent speech by the Chief of Staff General Gordon Sullivan at a STRICOM-hosted conference in May 1993 when he said,

> ...We will use simulations more to equip the force. We will determine our needs in large-scale simulation supported exercises to allow us to consider alternative solutions to meet our needs. Distributed Interactive Simulation holds great promise for compressing the acquisition cycle and saving money. Simulation lets us see and touch the acquisition cycle. I believe we can collectively help change our heel-toe system to one which is more cost effective and responsive...

STRICOM's mission includes the following:

- Technology base for simulation and training.
- DOD focal point for Distributed Interactive Simulation (DIS) environment.
- Acquiring training devices, instrumentation, threat simulators, and targets.
- Lifecycle sustainment support of fielded products.
- Quality support to the soldier.

The projects discussed later in this section are STRICOM's relevant efforts that focus on a next generation ITS.

5.2 Overall ITS R&D Program

The ITS program at STRICOM is managed and executed by the Directorate of Research and Engineering as part of the STRICOM Tech Base Program. STRICOM's ITS program is compatible with the AMC AI Master Plan (see Section 4.2 on page 51) and focuses its ITS efforts in two principle areas that closely support its mission: (1) ITS that incorporate existing simulations into the ITS architecture, and (2) ITS that are constrained

to include simulators and simulations as a component of the ITS design. The ITS development strategy addresses the ITS architectural and functional component design issues for the two areas, first in the context of individual training and then extending those results to the crew/team/combined arms training situations.

5.3 On-Going ITS-Related Projects

DOD's FY93.1 SBIR Program Solicitation included STRICOM's topic for "The Development of a Next Generation Simulation-Based ITS for Training Devices and Simulators." Subsequently, four Phase I contracts were awarded efforts (these efforts are described below). Funding for these Phase I efforts is provided jointly by STRICOM and ARPA/Advanced Systems Technology Office (ASTO). The remainder of this section discusses these efforts with status as reported in early 1994.

During Summer 1994, STRICOM, in coordination with ARPA/ASTO, will decide which project(s) to endorse for SBIR Phase II funding.

5.3.1 Simulator/Simulation-Based Intelligent Tutoring Systems Project

This Phase I effort has two research goals: (1) to develop an instructional framework based on models of expert problem solving that are suitable for incorporation in a simulation environment, and (2) to develop a simulation-based ITS.

Three innovations are being addressed in this work. First, instead of using the usual preprogrammed static scenarios, this system will regenerate scenarios dynamically based on diagnosis of student knowledge. This will allow the system to custom tailor scenarios to a student's level of learning. Secondly, the mode of student assessment will be based on elicitation techniques to determine the underlying levels of knowledge and problem solving. Finally, whereas ITS typically use a single representation scheme for knowledge, here an integrated approach using five different knowledge representation schemas will be employed.

Programmatic Background	This project is being conducted under SBIR Contract Number M67004-93-C-0083. The contract was awarded August 24, 1993, to the Research Development Corporation (RDC). The contract will terminate May 9, 1994. It provides a total of $98,695 in funds. Researchers at RDC will be supported by researchers from the University of Massachusetts and George Mason University. The Principal Investigator is Dr. John Leddo from RDC.
Planned Products	Proof-of-concept ITS for the subject area of trauma care.
Approach	Phase I consists of five tasks.

Task 1: *Testbed Selection.* The first step in the project is to select a testbed for the Phase I proof-of-concept system. Researchers will work with the client to select an appropriate topic. **This task has been completed. The testbed chosen is trauma care and the intended user community is Army 91B specialists, specifically those in the National Guard and Reserve. Course material for this topic is developed at the US Army Training Doctrine Command (TRADOC) in Ft. Sam Houston. Additionally, units such as those at Camp Dodge, Iowa have 91B's stationed and also participate in their ongoing training. As part of the project work, RDC has established a working relationship with both of these organizations. Researchers have met with instructors at TRADOC and will observe a training exercise in mid-February in which these skills are practiced.**

Task 2: *Developing an Instructional Approach.* The first step in the development of an instructional approach is to develop a model of the testbed knowledge. Since the goal is not simply to present a curriculum within the context of a simulation, but rather to build the same type of thinking skills in students that experts have, it is important to identify and model those skills. This model will then drive instructional requirements designed to build similar knowledge in students as well as drive the evaluation of how well students have learned that knowledge. These instructional requirements and evaluation feedback then drive what simulations and events the students experience. **This task is ongoing. Several initiatives have been taken. Subject matter experts to help with the knowledge and curriculum development have been recruited. Copies of training material produced at TRADOC have been gathered. The actual instructional framework will be completed shortly after the training exercise is observed. The knowledge modeling framework and assessment methodology are essentially complete; both of these center in RDC's integrated knowledge structure (INKS) framework that combines five separate AI representation schemes. These schemes are scripts, object frames, semantic nets, production rules, and mental models.**

Task 3: *Developing a Simulation-Based ITS Framework.* A successful simulation-based ITS is one that gives students practical problem solving experience in a realistic environment and, at the same time, effectively teaches students what they need to know. In Phase I, researchers will focus on what interactions are necessary between the simulator and simulations and the ITS to achieve this balance. They will focus on two key elements: how the ITS-based instructional framework is implemented in the simulator and simulation; and how the ITS collects feedback to assess what the student has learned and what additional instruction is necessary. **Work is partially complete. A general architecture exists that includes the following major modules: knowledge elicitor, knowledge assessor, lesson generator, and user interface. The details of these modules will be fleshed out once Task 2 is complete.**

Task 4: *Developing a Proof-of-Concept ITS.* Researchers will develop a proof-of-concept ITS on a 486 Macintosh. **The work is partially complete. The basis for the proof-of-concept ITS will be a previously developed cardiac tutor built by Dr. Woolf. Currently, the cardiac tutor has been "gutted" of the cardiac knowledge base.**

The trauma knowledge will be added once complete, as well as the scenario generation algorithms. Additionally, RDC's assessment approach will be added. The INKS framework will also be added to the cardiac tutor.

Task 5: *ITS Evaluation.* The researchers propose to evaluate the ITS with relevant Army subject matter experts and/or instructors, if possible. The ITS will be evaluated informally as to how well it presents relevant concepts. If possible, researchers will also have students review the ITS. In particular, they will be interested in how informative the students feel the system is. Army personnel at TRADOC and Camp Dodge have agreed to evaluate the ITS and provide interim feedback.

Potential Phase II & III Follow-On

Of the research is successful, this project could set a new direction in the education and training software fields. Additionally, resulting tutors could be used in Government training programs. Consequently, whereas Phase I will test the feasibility of embedding a simulation-based instructional framework in an ITS, Phase II will develop and test a full-scale simulation-based ITS. In Phase II, researchers will conduct a more thorough evaluation in a classroom setting to see whether the ITS enhances classroom performance as measured by the rate at which subject matter is mastered and overall problem solving performance. For Phase III, a joint venture with Innovative Thinkers, Inc. is planned to market the results of the proposed work.

5.3.2 Trauma Tutor

Development Status

Proof-of-concept prototype under development.

Architecture

As discussed in Section 5.3.1, Task 3, the goal of this project is to develop a generic architecture for building simulation-based ITS. Figure 17 illustrates the general architecture design. The two key components are discussed below.

Knowledge Elicitor

The first logical step in the ITS approach is to assess the student in terms of what he knows so that instruction can be tailored to meet his specific needs. The Knowledge Elicitor module performs this function based on hybrid knowledge elicitation technique. The Knowledge Elicitor also uses the existing Student Model or Student INKS in order to determine what gaps still remain in the student's knowledge.

To make the assessment as seamless as possible with respect to the training environment, the Knowledge Elicitor should have built-in knowledge regarding the learning environment, particularly the scenarios being used. Additionally, assessment can often be used in an instructional capacity. For example, a student can be asked to articulate his reasoning. Not only can this be useful in understanding how a student reasons, but it can compel students to examine their reasoning and learn from it. For this reason, the Knowledge Elicitor may use knowledge of Teaching Techniques in order to perform the assessment.

The Knowledge Elicitor takes data from the Student INKS as input, turns this into questions to present to the Learner (student), takes student answers, and passes them to the Knowledge Assessor.

Currently, the Knowledge Elicitor is largely complete as a generic processing module. What is missing from the development is the ability to integrate teaching techniques and scenario-based information. This will be pursued in future work.

Knowledge Assessor

The student's answers to the questions that have been presented are input into the Knowledge Assessor. The Knowledge Assessor module updates the INKS model of the student and then compares this student model with an Expert INKS that represents the knowledge an expert might have. Any discrepancy between student and expert would then drive the instruction that the student would receive. Additionally, the Knowledge Assessor updates the Student INKS model based on its assessment of how the student responded to the queries.

The Knowledge Assessor is partially complete as a generic module. The INKS structures themselves are well defined and have been used in a variety of systems. The researchers are currently developing an INKS editing tool that would allow users to build their own expert model.

Evaluation Status

Evaluation of the system is planned for mid-May 1994 at Ft. Sam Houston, and late May 1994 at Camp Dodge.

Operating Environment

The system runs on a Macintosh LC II. It is programmed in Macintosh Common Lisp.

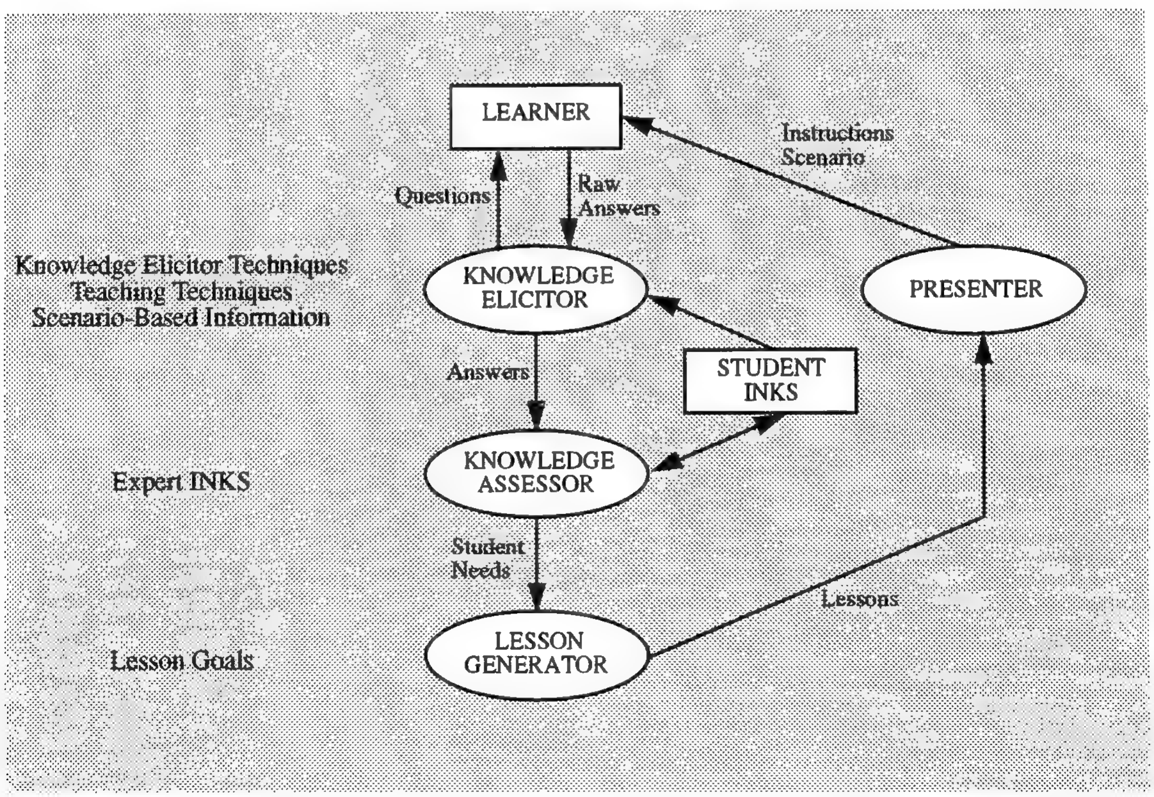


Figure 17. Architecture of the Trauma Tutor

5.3.3 Case-Based Reasoning for Simulation-Based Intelligent Tutoring Project

The goal of this research is to prove the feasibility of a software solution to providing intelligent training. Researchers will design a generic simulation-based ITS. The feasibility of the system will be proven through implementation of a proof-of-concept prototype in the Army medical specialist domain.

Because students often learn best by example, this effort will design a tutoring system around example problems. The examples are presented as simulated examples or simulated training exercises. These examples consist of a problem, its solution, the problem-solving process, and an explanation as to how that solution was derived. Teaching will be accomplished through the presentation of appropriate examples. Testing will be performed by presenting the problem part of an example and comparing the student's solution to the stored solution.

The major innovation in this work is the application of the AI technique of case-based reasoning (CBR) to capture and represent the example problems (or cases) and provide explanations to students on problem-solving techniques. CBR offers enormous benefits compared to standard AI approaches. The knowledge elicitation bottleneck is largely circumvented. By supporting automatic acquisition and presentation of domain knowledge for teaching (that is, the examples and the principles and procedures needed to solve them), a CBR-based ITS would greatly decrease the cost of developing new training software for the government and private sector. CBR also permits a system to represent multiple problem-solving methodologies. The system will be sensitive to a student's knowledge of problem-solving principles and tailor the teaching sessions accordingly. The body of knowledge contained in the system could be expanded over time simply through addition of more problems. An ITS based on CBR as described here would be best equipped to decide which simulations (examples) should be presented or trained and when.

Additional innovations include capturing the solution process to support explanation facilities, and supporting system "learning."

Programmatic Background

This project is being performed under SBIR Contract Number M67004-93-C-0084, SBIR Topic A93-022. The contract was awarded to Stottler Henke Associates, Inc. (SHAI). It began in August 1993 and is due to terminate in May 1994. The total award amount is for Phase I is $70,000.

SHAI will be supported by information received from domain experts in the US Army Academy of Health Sciences; ARPA Support Team; Camp Dodge, San Mateo County, CA; Stockton, CA; and the American Heart Association. The Principal Investigator is Mr. Richard Stottler from SHAI.

Planned Products Proof-of-concept ITS for emergency medical procedures. Design for a Phase II prototype.

Approach An eight-task approach is proposed for developing the techniques and implementations associated with the Phase I objectives.

Task 1: *Develop the specific trial problem domain for a CBR trial application.* Working in conjunction with Army representatives, at least two separate problem solving areas will be identified as possible teaching applications and one will be chosen in Phase I to demonstrate the CBR approach and the implementation of a case-based system. **This task has been completed. Army Medical Specialist and Tank Battle Tactics were selected as possible teaching applications. The Army Medical Specialist application, more specifically, the AMEDD NCO Level 2 91 B Technical Training, has been chosen in Phase I to demonstrate the CBR approach to ITS.**

Task 2: *Define the Preliminary Case Structure for the Elicitation Procedure.*Working with the specific trial problem, researchers will determine an appropriate representation for cases in the subject area. The cases will be presented as simulations and will likely consist of attributes relating to the simulated scenario as well as problem-solving principles and methods. Researchers will also begin examination of potential similarity metrics and retrieval methods. **This task has been completed.**

Task 3: *Conduct Application of CBR Elicitation for the Purpose of Developing Case Histories.* In cooperation with domain experts, SHAI will elicit cases and other knowledge required for the CBR ITS. Interviews (of approximately one hour each) with the selected domain experts will be held. Interview materials will be constructed consisting of (1) description of target case (a particular example), (2) description of the comparison case, (3) the possible similarity measures, (4) the set of principles or methods used to solve the problem, and (5) the known principles and methods in the comparison case. The domain experts will be interviewed individually and asked to solve the problems in their usual manner, supplying explanations for individual steps. Each domain expert will be asked to (1) verify the principles and methods checklist, (2) identify differences between the target case and the comparison case for each principle and method, and (3) generate specific problem solving approaches. **This task has been completed.**

Task 4: *Develop/Design the Structure for the Case Base.* Once the knowledge is collected, a refined structure for representation must be developed. This structure will drive and also describe the retrieval and processing methods. The case structure must be capable of representing an example which includes the problem, its solution, and an explanation of the solution which will reference general principles or methods. An object-oriented approach is viewed as best suited for case representation and researchers will use object-oriented programming techniques to develop an integration framework. This will provide the capability to instantiate a given domain tool as a resource object in the system from which messages can relay various levels of program assistance. Both a case structure and a structure for the overall case library are needed. The case structure must consider the various retrieval processes to be used in Task 5. In addition to refining the preliminary case structure for the trial domain, the general domain-independent case structure will be refined, if

required, so that the trial domain remains a specific instance of the general case structure. **This task is in progress, knowledge engineering has begun. The task is being performed concurrently with Task 5.**

Task 5: *Develop/Design the Retrieval Process and Reasoning Structure for the System.* A retrieval method will be outlined for the system as the case structure is defined. Classification of similar cases into clusters is one method to determine the most similar match based on compared attributes from the user's present situation. Other rapid retrieval methods include the use of feature weighting and proximity measures. Either a hierarchical-domain-space classification and retrieval method or a combined quantitative/qualitative rapid retrieval method may be used for this CBR system. Once the cases are retrieved, reasoning functions can then operate on the information. Reasoning functions include the following:

- Presentation of the stored solution methods from the retrieved case to the user (if one case was found that directly matched the user's needs).
- Presentation of the stored solution methods from the retrieved case to the user (if one case was found that directly matched the user's needs).
- Reasoning upon a set of similar cases to provide a common solution.
- Reasoning with hypotheticals. Hypothetical cases provide for extensions to the existing set of cases by providing bias on identified attributes. Hypothetical cases provide a biased view of a given situation and provide more comparison cases for a situation.

It is the intelligent retrieval which serves as the primary driver of the course curriculum. Part of the retrieval process will be an assessment as to how well the selected case was evaluated previously. Every time a case is retrieved the student tells the CBR system whether the returned case was helpful or not. Thus the next time the system retrieves a case, given the same scenario, the previously returned unhelpful case will not be a part of the retrieved set (that is, the system "learned" from its previous mistake and will therefore improve with use). This method of learning is quite different from the usual improvements easily realized in a CBR system where new cases are added for an automatic improvement. **This task is in progress; it is being performed concurrently with Task 4.**

Task 6: *Implement the Prototype.* Based on the previous tasks, researchers will develop a small one-user proof-of-concept prototype. This prototype will provide a sample of the "look and feel" of the system and contain representative CBR functionality that operates on a subset of the intelligent tutoring domain. It will be used to demonstrate how a domain expert can enter knowledge via cases and how the system can automatically modify the presented material to meet current student needs.

Task 7: *Assess the validity of the prototype.* To prove the feasibility of this effort, the validity of the prototype must be evaluated. Does the system represent examples adequately? Are the retrieved similar cases useful to the students? Will the system be accepted by student and teacher? These are questions to be addressed by the validity evaluation.

Task 8: *Prepare the Phase II Design and Final Report.* The final report will describe the development and architecture of both the general and the specific case structures and retrieval methods and will include the Phase II design. This design will include the architecture for all modules. Automated methods of knowledge elicitation for subjects will be discussed. The evaluation of the prototype in its trial domain will be presented. A future research section will outline the requirements needed to develop a general intelligent tutoring prototype, applicable across many subjects. This prototype would serve as a starting point for a commercial system.

Potential Phase II Follow-On Phase II tasks will address the implementation of both the full-scale General ITS development capability and the Medical Specialist ITS. This will include knowledge engineering; software design; and incremental and iterative software development, testing, and evaluation.

Medical Specialist After feasibility has been proven by the Medical Specialist ITS prototype demonstration in Phase I, SHAI will build the full-scale ITS in Phase II with SBIR funding. It will then be ready for operational use in the Army's 91B course. Only minor modifications will be required to make the system applicable to civilian paramedic training. This represents a very large market potential since paramedics are trained throughout the country, primarily by individual counties. SHAI is in contact with a number of the Emergency Medical Departments of various counties who have shown great interest in this project.

Generic ITS Capability During Phase I an ITS development capability will be designed to facilitate the creation of Simulation and CBR-based ITSs. This design will be implemented along with the Medical Specialist ITS with Phase II SBIR funding. After Phase II this capability can be used to easily develop ITS for other domains. For each domain, approximately $10K of effort would be required by SHAI to develop an ITS prototype for demonstration purposes. Additionally, the development capability will be commercialized to create an authoring tool which allows non-programmers to develop simulation-based ITSs. This authoring tool could be mass-marketed. The authoring tool would be useful in the same domains as described above.

5.3.4 Army Medical Specialist ITS

The Medical Specialist ITS provides instruction on emergency medical procedures. It provides the student with both the knowledge required to physically perform the procedures and also the cognitive skills required to determine when it is appropriate to perform a particular procedure.

The final system will be able to automatically query the subject matter expert, or teacher, automatically. It would query the subject matter expert for two different things: (1) examples, and (2) the principles and methods the students need to learn. The subject matter expert could order these principles and methods in the chronological order the students should learn them. From this information, the CBR system could automatically design a

course of study. Each related set of principles and methods would form a section. At the beginning of each section, the system would present a motivating example in hypertext form. Next, the principles and methods would be presented, followed by more hypertext examples. Finally, at the end of the section, testing would occur with the presentation of just the problem part of examples which the student has not yet seen, but which use the principles and methods from the section. The student's answers (and method of solution) would be compared to the stored solution. The CBR system should also be able to diagnose the student's progress based on which problems were missed and take remedial action. If certain principles are common to the missed problems, more examples using those principles could be presented.

Such a system would allow individualized instruction, letting students learn at their own pace and in their own way. By getting student feedback on the examples and problems presented, the system could learn from its mistakes to improve its performance. The addition of more cases would also improve the system; one way of accomplishing this is to allow correct student solutions to become part of the case base. Such as system would reward and propagate original creative thought and problem solving.

Development Status

Proof-of-concept prototype under development.

Architecture

The Medical Specialist ITS is based primarily on a simulation of the human body and a data base of scenarios called the case base. The ITS will employ a modular object-oriented architecture so it can be reapplied to different domains with a minimum of effort. The modules include instructional knowledge, simulation/trainer knowledge, domain knowledge, tool knowledge, diagnostic knowledge, student models, user interface, a blackboard, and message passing modules. The overall architecture is depicted in Figure 18 on page 75.

Instructional Knowledge

The instructional knowledge specifies how the student should be taught. It provides an initial structure for a course of study by defining the order of presentation of procedures and principles. For each procedure and principle, an illustrative example is retrieved and shown through the simulation. The student may retrieve additional examples which illustrate the points, thus tailoring the course for themselves. At certain points in the course, the student is tested by retrieving and simulating medical scenarios. The student's performance is recorded in the Student Model. From this information, the Diagnostic Knowledge module deduces the procedures and principles that the student does not understand, and different examples illustrating them are retrieved and shown. Other exercises which make use of such procedures and principles are used to retest the student. In this way a remedial course of study is tailored to the individual.

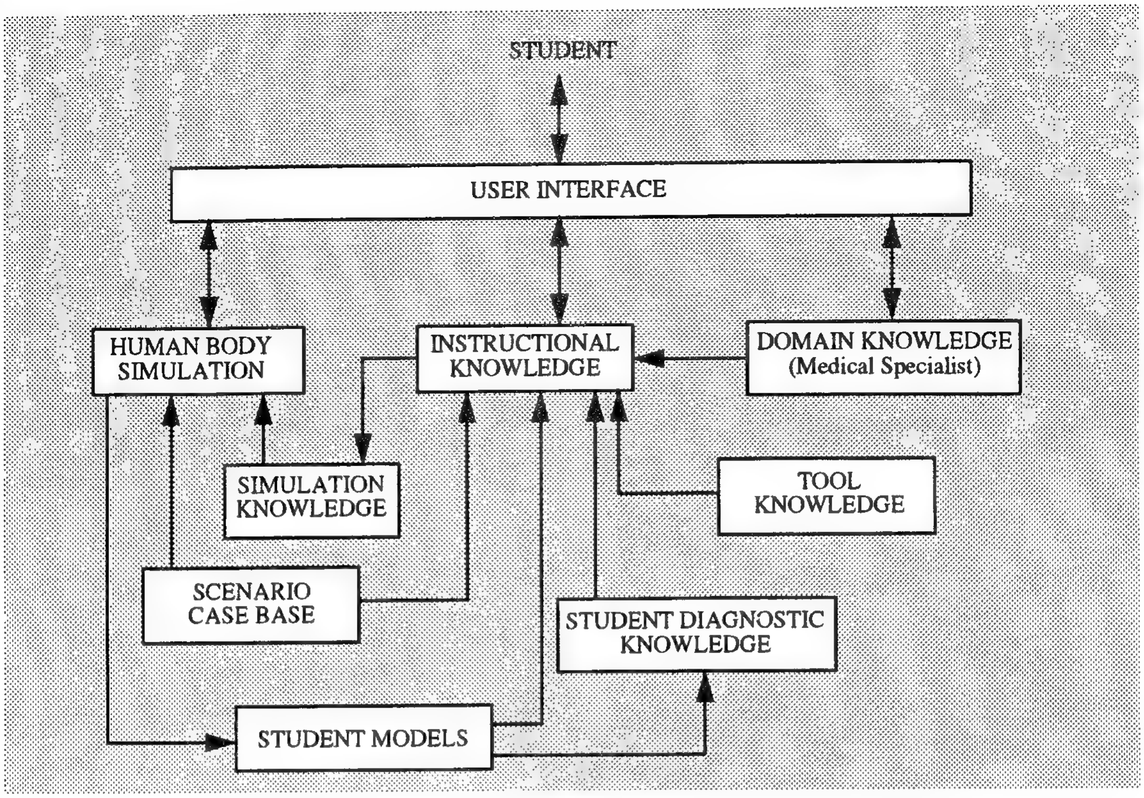


Figure 18. Army Medical Specialist ITS Architecture

Simulation/ Trainer Knowledge This module contains information about the simulations and trainers available to the student. The information is used to generate scenarios for the simulations, to run the simulations, and to possibly control some of the entities in the simulations.

Domain Knowledge This module contains examples and problems in the form of cases that contain the problem description, the solution, and the problem-solving process with explanations. The domain knowledge also includes principles, procedures, techniques, and rules used to solve problems. These are referenced by the explanations of the solution steps. The solution process may consist of steps which are serial or parallel in nature, thus providing the student with some reasonable flexibility in the order he decides to tackle his portion of the problem. Knowledge on how to present the examples and problem solving methods may exist here as well.

External Tool Knowledge In many domains there are existing tools which must be used in problem solving. These tools may or may not reside on the training computer. They include software, manuals, handbooks, diagrams and drawings, files, and special equipment. The ITS must posses knowledge of these tools in order to know when and how they should be applied.

Diagnostic Knowledge This module allows the ITS to diagnose the student's deficiencies. It looks at what steps of what problems the student has erred in. The errors include missed concepts, misapplied actions, missed opportunities, and incorrect classifications.

Student Model The student model will contain the student actions and decisions from different exercises; the principles, procedures, and techniques which have been presented; and those that have been mastered based on performance of exercises. The set of principles, procedures, rules, and tools referenced in the solutions of problems the student has solved successfully represent the student's acquired skills. The student model will be high fidelity and reflect the skills, knowledge, and error-rate of the student.

User Interface When designing an ITS is it important to maintain flexibility as to how information should be presented. This prototype's media include animated simulations, high resolution pictures, video footage, hypertext, as well as more standard forms of text and graphics. The course is arranged in a hierarchical structure of topics. The student is free to navigate this hierarchy, viewing the course material and examples in any order.

Blackboard and Message Passing These modules are required in team domains to allow flexible communication and distribution of problem information between nodes. Some domains will best represent communication between participants as a blackboard; this would be the case where everyone is in the same small room: anything anyone says is heard by everyone. In other domains, communication is represented better by message passing; this is the case where participant are geographically separated and phone calls are used for communication. Many domains may use a mix of these models.

Operating Environment MicroSoft Windows.

5.3.5 Hybrid Simulator-Based Intelligent Tutoring System Project

The objective of this SBIR effort is to develop a generic simulator-based ITS architecture and set of component module specifications. A primary design goal is to ensure that the architecture will support ITS upgrades of existing simulators as well as development of new ITS training devices. The approach works within the conventional ITS framework of explicit agent representation (teacher, student, expert), and combines skill-based adaptive training techniques used in ARI's Intelligent Flight Trainer project (see Section 7.2.2 on page 105) with the structured knowledge-based diagnosis and syllabus modification used in MICOM's Intelligent Embedded Operator Assistant program (see Section 4.2 on page 51). The overall architecture incorporates key features of both approaches with an emphasis on closed-loop operation of the man-machine trainer. Explicit cognitive models are used to represent both student and expert, with several model "sites" reserved for capturing student and expert differences. Components of the tutor model are implemented via algorithmic, neural network, or expert system approaches, depending on the component requirements and the enabling technology capabilities. The prototype implementation will employ CASYS, an object-oriented graphical simulation language that can support the proposed hybrid solution.

Programmatic Background

Charles River Analytics, Inc. has been awarded DOD Contract M67004-93-C-0085 to perform this work. This contract supports a six-month feasibility demonstration, with expenditures not-to-exceed $50K. The Principal Investigator is Dr. Greg Zacharias.

Planned Products

Proof-of-concept generic simulation-based ITS with an application domain in medical technician training. A set of component module specifications. Requirements specification for a full-scope Phase II prototype.

Approach

The proposed effort consists of four tasks.

Task 1: *Development of Hybrid Architecture.* Development of a hybrid ITS architecture to support simulation-based training. Review of any additional ITS and simulator designs relevant to the effort, to identify key features not accounted for in the proposed hybrid architecture, and to modify the baseline design accordingly to incorporate the desired features. The modified baseline will then serve as the preliminary design specification. **This task has been completed.**

Task 2: *Specification of Component Modules.* Development of functional specifications for the component modules composing the hybrid architecture developed under Task 1. Evaluation of functional implementation requirements against the capabilities of several enabling technologies, including cognitive operator modeling, artificial neural networks, and expert systems. Following the selection of the most appropriate technology, development of the functional specifications for each module, at appropriate levels of detail for the prototyping effort. **Candidate technologies for component modules have been evaluated. This task has been completed.**

Task 3: *Implementation of Prototype ITS.* Implementation of a prototype ITS following the Task 1 specification of the hybrid architecture and the Task 2 specification of the component modules. Implementation will be in the graphical object-oriented language CASYS. A non-real time prototype will be developed and tested against a student model to evaluate feasibility of the overall hybrid approach.

Task 4: *Requirements Specification for ITS Development.* Specification of requirements for Phase II development of a full-scope real-time prototype. Results of the Task 3 demonstration will be used to guide the requirements specification. A final report will summarize the Phase I findings and Phase II recommendations.

Potential Phase II & III Follow-On

The Phase II effort will use the Phase I requirements specification to develop the prototype ITS. The Phase II effort will focus on (1) stand-alone prototype development and validation, (2) integration with the selected training devices, (3) demonstration of real-time operation, and (4) evaluation of effectiveness relative to conventional usage of the training device. The results of the Phase II effort will provide the foundations for a Phase III product development effort aimed at bringing to market a stand-alone programmable ITS module targeted for integration with a number of existing training device simulators.

The Phase III effort will focus on developing a generic and flexible stand-alone ITS module, along with an interfacing protocol for upgrading existing systems. Demonstration and evaluation of this package will be guided by the Phase II pro-

totype evaluation experience, so that a critical early evaluation of the Phase II product can be made before significant development effort is expended. In the Phase III work, developers will also explore the commercial viability of direct integration of the ITS module with future training simulators, but the current assessment is that the add-on after-market potential is significantly larger.

Developers see a number of potential Federal Government applications for a generic simulator-based ITS, particularly in the areas of military flight simulators, naval nuclear and conventional power plant trainers, and trainers for personnel at Terminal Area or Enroute Air Traffic Control. Potential applications in the commercial area include training in rail traffic management centers, power plant and chemical process centers, and power systems dispatch centers. Commercial benefit in the potential spin-off of the systems development language CASYS is also envisioned.

5.3.6 Hybrid Simulator-Based Intelligent Tutoring System

Development Status Proof-of-concept prototype under development.

Architecture The proposed ITS architecture for simulator-based training is illustrated in Figure 19 on page 79. Horizontal dashed lines separate the figure into three major components: a simulation component, a student model component, and a tutor model component. The following description of these components relate back to this figure. The discussions are given in terms of a hypothetical example used for illustrative purposes.

Simulation Component The simulation component emphasizes the essential closed-loop nature of the training task being performed by the student operator. The operator generates discrete and continuous control actions (button switches, joystick movements) which are picked up through the controls half of the human-machine interface (HMI) module. These controls then drive the simulation. In a relatively passive monitoring task, such as the monitoring portion of an Air Defense Task, for example, these controls will primarily affect the choice of sensor suite, field-of-view, display gains, etc., and will probably have little effect on the combat environment in which the tactical system operates. In a more active weapons control task, such as target designation, tracking, and launch, these control activities will (should) eventually drive the simulated external threats, to take evasive actions, to use countermeasures, to counter attack, etc. The result of these changes in system simulation state are shown being fed back to the display half of the HMI, for presentation to the student operator. The closed man-machine loop shown here thus emphasizes the active effect the student operator has on the system.

Student Model This component focuses on the interactive identification of the key student characteristics, as embraced in an on-line model of that student. A second closed loop is shown beginning with a student model generating model responses in reaction to the system displays presented on the HMI. The student model responses are then compared with actual student responses by a model mismatch evaluator. This module calculates an error metric proportional to the disparity between student and

model, and adjusts the model accordingly, via the feedback path shown. Interactive adjustment of the model serves to minimize the disparity between student and model. The underlying hypothesis here is that the student model is then congruent with the student, at least on the operational dimensions of the simulated task at hand. In effect, the student model encapsulates the student's knowledge base of the system, and the student's skill level for manipulating system knowledge.

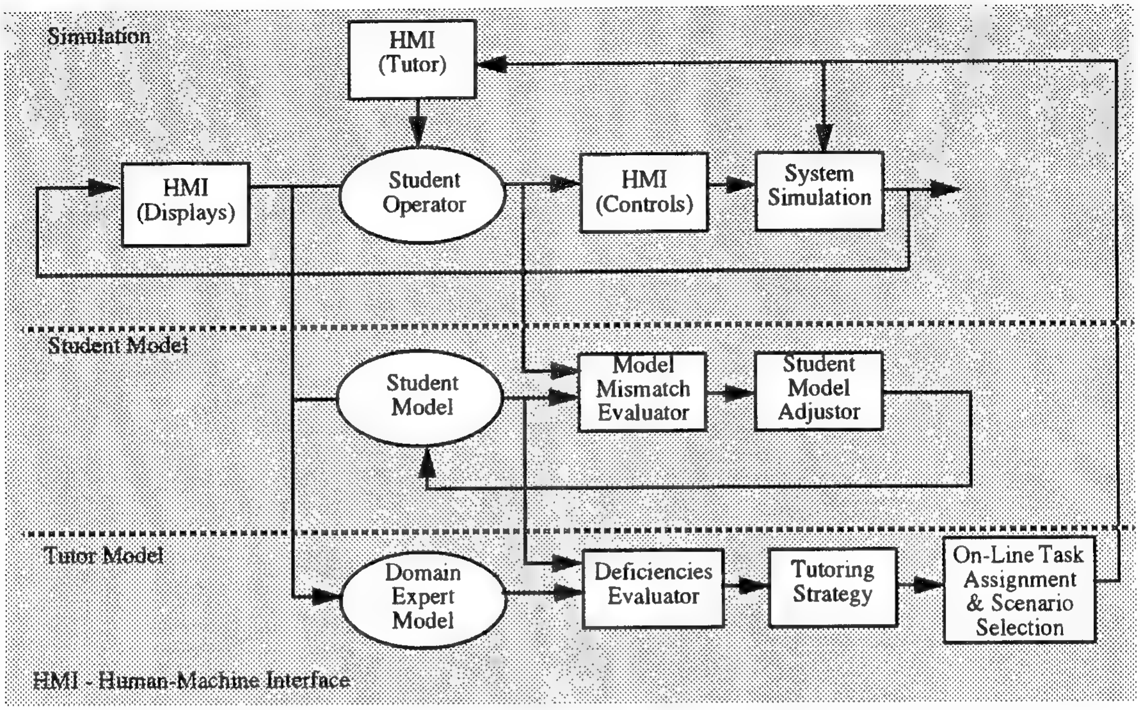


Figure 19. Simulator-Based Training ITS Architecture

Tutor Model This component focuses on the ITS pedagogical response, given its inferred knowledge of the student operator. Here the architecture shows an information path which begins with a domain expert model (that is, an expert operator of the system) generating responses in reaction to the system displays presented on the HMI, and presented to both the student and student model. The expert model responses are then compared with the student model responses, by a deficiencies evaluator which determines the disparity between student and expert models. Several techniques can be used here:

- On a skill-based level, deficiencies can be compared with the response from the optimal control model of the operator.
- On a rule-based level, they could be evaluated in terms of appropriate or inappropriate rule-firing.
- On a knowledge-based level, they could be evaluated in terms of overlay coverage of genetic graphs of student knowledge.

Whichever approach is taken, and several are likely to be necessary to assure coverage at the different cognitive operator levels, the identified deficiencies then drive a tutoring strategy. This specifies the scenario difficulty level, the type of task(s) to be engaged in by the operator, any training aids that might be used, and any advice (that is, coaching) to be given the student. This particular selection, for the next run or series of runs, is then fed back to both the student and the system simulation. The student may receive a critique of his past performance, advice on how to remediate shortcomings, anticipatory information regarding the upcoming scenario, advice on how to deal with a particular aspect of it, and other guidance. This is presented to the student through the third portion of the HMI, that portion which is ITS specific and not necessarily tied to the simulator displays. In parallel, the system simulation will receive information needed to, for example, change the scenario or adaptively adjust difficulty levels and tempo to reflect the tutoring strategy objectives.

Operating Environment

Under Phase II, the researchers plan to expand the hybrid capabilities of the proposed system by conducting the development in CASYS, which is currently under development by Charles River Analytics Inc. CASYS has been chosen since it will support real-time algorithmic, neural network, and expert system representations of individual components of the proposed ITS. In addition, CASYS will allow development of a general hybrid environment for the overall system, to support an integrated approach to a hierarchical and modular system specification.

CASYS incorporates a system description language to support the specification of application knowledge for complex systems and the development of an intelligent system based on this specification. Complex system models combine topological knowledge (interconnections and hierarchies among components) and procedural knowledge (functionality of components). CASYS is capable of representing both the topological and procedural aspects of a system model. CASYS has developed specifications that can be compiled directly into Ada programs on a variety of platforms.

5.3.7 Icon-Based Intelligent Tutoring System Utilizing Fuzzy Expert Systems and Multi-Media Gaming Simulations Project

The overall goal of this Phase I research program is to demonstrate the feasibility of implementing expert system information processing techniques and fuzzy logic paradigms on multimedia-based desktop computer systems for application to the development of an ITS capable of analyzing user performance and interest parameters and providing an intelligent learning strategy based upon gaming simulation techniques. The project seeks to demonstrate that ITS can be taken from the laboratory and used to both teach low-density skills and extend basic training.

The major innovation is an expert system control architecture containing fuzzy logic sub-modules interacting with multimedia sources to generate an intelligent tutoring envi-

ronment. The goal of expert system analysis and control techniques is to capture perfected skills and provide a comprehensive teaching experience based on these skills. The fuzzy logic allows for situations where student response is not a simple yes or no, enabling the student to give a fuzzy response (for example, one that is fairly close to the right solution) and so allow more accurate appraisal of student capabilities. The effort will demonstrate a proof-of-concept system which uses fuzzy expert system analysis and control techniques and multi-media presentation techniques for application to intelligent tutoring and associated gaming simulations. The results anticipated include the development of a fuzzy expert system architecture and graphical user interface which is flexible enough to be used in a variety of complex tutoring and gaming applications.

Programmatic Background

This project is being conducted under SBIR Phase I Contract, Number M67004-93-C-0082, awarded to the American Research Corporation of Virginia (ARCOVA). Begun August 1993, the contract expires in May 1994. The total funding awarded under the SBIR contract is $49,930.

The research effort is being conducted by a development team consisting of investigators employed by ARCOVA and consultants from Radford University and Virginia Polytechnic Institute and State University. The work is being conducted in the ARCOVA laboratories. The Principal Investigator is Mr. John Neal from ARCOVA.

Planned Products

Proof-of-concept prototype with two data bases, that is, two application domains: (1) 63 Bravo, five-ton truck oil system repair, a low-density specialist task, and (2) 91 Bravo, advanced first aid skills to extent basic training.

Approach

The Phase I research approach involves the design and evaluation of the expert system controller, fuzzy expert system analysis modules, the data base subsystem and the multimedia user interface software and hardware for the Phase I proof-of-concept system for application as an ITS capable of analyzing user performance and interest parameters and providing an intelligent learning strategy based on gaming simulation techniques. The specific technical objectives for Phase I are presented as a sequence of events constituting the research approach.

Task 1: *Design of Learning Methodologies and Supporting Instructional Strategies and Gaming Simulations Which Can Be Implemented on Standard Desktop Computer Systems, and Development of Algorithms and Correlated Input/Output Data Base Arrays and Rule Sets for the Implementation of the Designed Learning Environment.* This technical objective involves the evaluation of learning methodologies to ascertain instructive procedures which can be successfully implemented on standard desktop computer systems. This evaluation is used to design a tutorial strategy and associated gaming simulations for selected instructional topics. **In the area of database development, work has been accomplished on obtaining data through the identification and acquisition of manuals on maintenance of five-ton trucks (Military Occupational Specialty 63B) and manuals on the emergency medicine and ground ambulance evacuation support (Military Occupational Spe-**

cialty 91B); the definition of data base item formats; and the specification of expert system database access and control. The selection of MOS's 63B and 91B as curricula for the ITS development was driven by the STRICOM and ARPA requirements defining the need for ITS training for National Guard and Reserve Forward Support Battalions. ARCOVA is currently making arrangements with local National Guard and Reserve units to video tape and photograph selected maintenance tasks and medical procedures relating to MOSs 63B and 91B.

Task 2: *Design of an Interactive Icon-Based Multi-Media User Interface.* This technical objective involves the design of a flexible interactive icon-based diagrammatic information input, retrieval, and presentation environment based on multimedia presentation techniques capable of addressing the needs of users with varying information access and display requirements. Use of the interface is based upon a windows and menu-option format with an icon-based diagrammatic representation of topics and associate gaming simulations. **Work has been accomplished on the specification of the interface features and the structure of speech recognition capabilities (COVOX standard).**

Task 3: *Development of a Control and Analysis Expert System Architecture Which Supports Fuzzy Expert System Preprocessors.* This technical objective involves the specification and design of an expert system architecture able to accept analysis results from fuzzy expert system preprocessors and control a human-computer interface. The completion of this objective also entails the design and development of the fuzzy expert system preprocessors. **Work has been accomplished on the implementation of the inference engine and the specification of the user model architecture, specifically the human-performance variables, including detection of task arrival, performance accuracy, and performance time.**

Task 4: *Expert System Testing and Verification.* This technical objective comprises the performance testing and evaluation of the Phase I proof-of-concept system. The testing results will provide an assessment of the expert system and user interface performance including accuracy, speed, and ease-of-use.

Task 5: *Evaluation and Optimization of a Proof-of-Concept System.* The software and hardware designed and implemented in the previous technical objectives will be integrated into a complete Phase I proof-of-concept system that will be tested, debugged, and evaluated. System performance will be optimized to meet the Phase I design specifications.

Potential Phase II & III Follow-On

The proof-of-concept ITS architecture developed during the Phase I program will demonstrate the major features of the ITS concept being developed. In Phase II, it will be expanded to develop a fully featured laboratory prototype. The Phase II system will be a prototype implementation of technology that can be transferred to the US Army and ARPA. The Phase III program will involve the translation of the Phase II laboratory prototype to a production model for publication on CD-ROM media.

The intention is to commercialize the results of the Phase II research and development effort during Phase III of the program. ARCOVA plans to bring the proposed technology to the marketplace through a combination of venture capital and com-

mercial funding as a result of follow-on-funding commitments. It is anticipated that commercialization will initially occur in Federal Government organizations with potential customers including the US Army and other components of the US Department of Defense. Commercialization within the Federal Government will then be followed by a transfer of the technology to the private sector, beginning with the selection of a marketing firm through an evaluation of existing marketing capabilities and compatibility with commercialization of the proposed product.

5.3.8 Icon-Based Intelligent Tutoring System

A three-mode integrated tutoring methodology will provide a sequence of learning stages. The first stage of the learning process will be tutorial presentation of concepts, conducted in parallel with drill and practice sessions. This stage will be based on a fairly short timeframe, requiring complete student attention and retention. The second stage of the process will be simulation and will teach topic operational scenarios in a tightly controlled operating environment. This stage will require less user attention and retention than the previous stage and will allow the user to relax while becoming familiar with operating procedures. The last stage of the process has no fixed timeframe and uses interactive educational gaming that allows the user to operate freely within a virtual world environment. This stage will require varying degrees of user attention and retention depending on the gaming situation. The student will be assigned goals and monitored on various performance parameters as the goals are achieved. The gaming simulations will be based on competition with either the computer or with other users, depending upon the characteristics of both the topic and the classroom, and will be dynamically re-configured during the gaming session as areas of student knowledge that need improvement are identified. This dynamic re-configuration will tend toward gaming environment goals which emphasize the areas in which the student has not demonstrated proficiency.

The CD-ROM-based ITS concept being developed would store both the multimedia database, the expert system modules for intelligent tutoring, and reference and record entry graphical user interface modules on CD-ROM. Once trained, a soldier could carry the CD-ROM which was used during his MOS curriculum into the field for on-site reference. The CD-ROM would be more portable and more rugged than a complete "hard copy" manual set and, when accessed through the intelligent reference interface, would provide more accurate and more efficient information retrieval. In addition, the soldier could carry a set of floppy diskettes which would be used to digitally record "paper work" in the field. The intelligent record entry data base interface would provide access to paper-form templates which can be completed and stored on floppy diskette or printed out as

"hard copy" in the field. The intelligent form generation function would provide example entries and suggest correct responses to form sections.

Development Status Proof-of-concept prototype under development.

Architecture The overall architecture of the system is shown in Figure 20 on page 85. The hierarchy of the intelligent tutoring system, when viewed at the top level, can be divided into three major subsystem classes: the multimedia database, the expert system/ fuzzy logic analysis and control subsystems, and the GUI subsystems.

Multi-Media Database The development of the multimedia database involves the conversion and integration of several information sources. This process entails the acquisition of a comprehensive information source set as related to a given subject. Such sources may include software gaming simulations which model procedures and skills, training and reference manuals relating to the subject, and photographs and video clips pertaining to selected subject topics. The information is stored within the multimedia database in a format which can provide rapid, efficient access to selected items when referenced by the expert system controller. The database represents the core knowledge of the selected subject. It is being designed as a modular system which can be easily updated with new information sources or modifications of existing data items. This will allow the efficient and accurate revision of existing manual items and software simulations as updates are issued.

Expert System/ Fuzzy Logic Analysis & Control Function The expert system/fuzzy logic analysis and control function has one purpose: to monitor and control the user input/output information feedback path. This function represents the most complex part of the ITS development effort. The expert system maintains an information flow feedback path which encompasses user interface control information, an audio/visual information path between the computer display and the user, and user input response information path returning to the expert system. To maintain correct, accurate, and efficient control of this path, the expert system interacts with four major data storage elements: the multimedia data base discussed previously, a fixed expert rule set and data array, a modifiable expert rule set and data array, and a user model and personal characterization data base. This modular design will allow the base expert system architecture to be used across a wide variety of courseware subjects and information sources without significant modification.

User Interface The ITS user interface is being designed for maximum ease-of-use while maintaining accurate, efficient data base access. Most of the interface functions are accessible through the selection of iconic elements and all major interface features are graphics based. Efforts on the Phase I program are being directed toward the development of an intelligent tutor interface. This interface supports all information classes in the multimedia data base (video, still photography, manual text, and software simulations) and presents training courseware in the three learning mode sequences discussed previously. The user interface environment is structured as a window presentation in which each section of text, each table, and each figure is contained in a separate window.

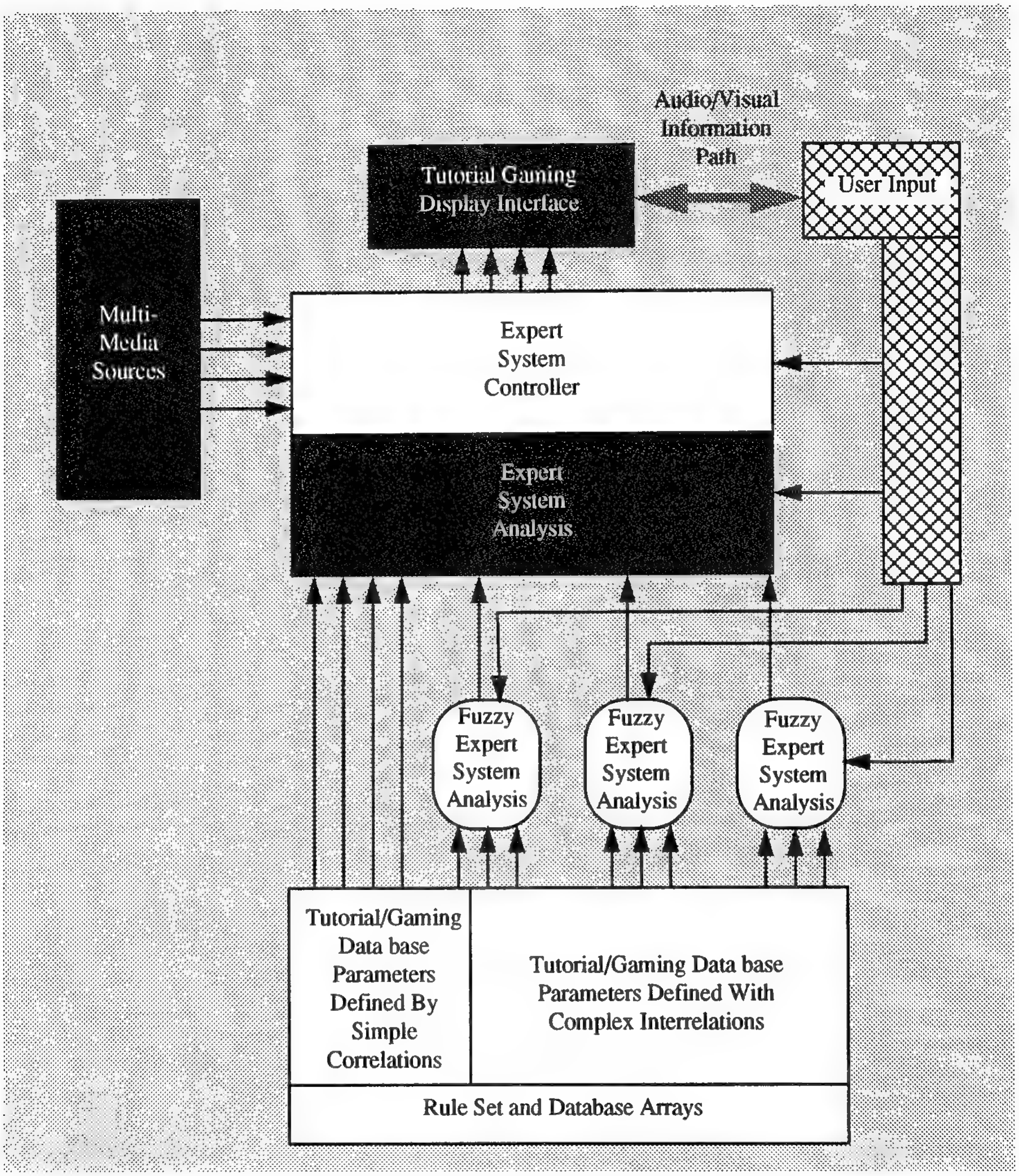


Figure 20. Icon-Based ITS Architecture

In addition to the expert system multimedia tutoring functions, the Phase I user interface will provide the student with several learning environment features. These features are designed to replicate the tools a student would normally need in a typical teacher and classroom environment. They include out-of-sequence reference links implemented as “hot spot” areas; searching for selected text blocks or words in text sections, figures, and tables; a zooming tool allowing a user to magnify or reduce text, table, or figure views; a notepad tool which allows a user to save personal notes in ASCII files which are individually linked to selected documentation elements; a book mark tool which allows a user to mark multiple items in the documentation for quick future reference; and the ability to select and print hard copy documentation of items or portions of items within the multimedia database.

Operating Environment

The Phase I proof-of-concept demonstration is being implemented to run under DOS on a desktop personal computer system. It is anticipated that the number of operating environments which will be supported by the ITS will be expanded in Phase II of the program to include Macintosh OS, OS/2 and Unix.

6. ARMY RESEARCH INSTITUTE, ADVANCED TRAINING METHODS RESEARCH UNIT

6.1 Mission and Role of ATMRU

The mission of the Advanced Training Methods Research Unit (ATMRU) has been stated as to increase the effectiveness and efficiency of developing and delivering training by the application of state-of-the-art computing technologies.

6.2 Summary of Past ITS-Related Work

Initially, ARI's work in the area of ITS was conducted by the Logistics and Training Technical Area and the Instructional Technology Systems Technical Area. These technical areas accomplished such notable achievements as the development of the MACH-III tutor. In the late 1980s, they merged to form the Automated Instructional System Technical Area. Notable work in this time period included the development of the INCOFT tutor. With ARI's change from the use of technical areas to research units, the Automated Instructional System Technical Area became the Advanced Training Methods Research Unit at the beginning of 1994.

In addition to developing specific tutors, ARI has been prominent in promoting an interdisciplinary approach to ITS development. For example, ARI has played a lead role in the conduct of the following workshops that have brought key researchers together:

- NATO's Defense Research Group Panel VIII, Spring 1985, dealing with the value of computer-based training in military environments [Seidel 86].
- NATO's Defense Research Group Panel VIII, Fall 1991, dealing with the application of advanced technologies to training design [Seidel 92].
- Intelligent Tutoring Systems for Foreign Language Learning: The Bridge to International Communication, 1991, dealing with second language learning [Swar 92].
- ARI Language Workshop, May/June 1993, also dealing with second language learning.

In addition, ARI researchers have prepared many reviews of ITS technology that have helped to promote an understanding of the changes in this field. Table 2 on page 88, for example, reproduced from Shute's ITS work [1994], shows how research focuses have changed over time.

Table 2. Important Issues Related to ITS Development

1970s	1980s	1990s
Problem Generation	Model-Tracing	Learner Control
Simple Student Modeling	More Buggy-Based Systems	Individual vs. Collaborative Learning
Knowledge Representation	Case-Based Reasoning	Situated Learning vs. Information Processing
Socratic Tutoring	Discovery Worlds	Virtual Reality
Skills & Strategic Knowledge	Progression of Mental Models	
Reactive Learning Environments	Simulations	
Buggy Library	Natural Language Processing	
Expert Systems & Tutors	Authoring Systems	
Overlay Models/Genetic Graph		

6.3 Overall ITS R&D Program

The ATMRU Automated Instruction Research program is driven by a conceptual model of the instructional development process. This model, shown in Figure 21 on page 89, provides a structured framework for "selecting learning and training research topics and issues in a total training R&D context, with the goal of producing systematic research-based guidelines for developing training programs and for ultimately evolving techniques to improve Army training" [Seidel 94].

The model allows relating R&D projects to four different stages in the development process. In the case of knowledge and skill learning, it shows that these issues need to be studied in the context of specific subject domains, and may address either the learning of individuals or that of a group. None of these elements is totally independent of all other elements. For example, understanding of the cognitive structure and problem-solving skills of subject matter experts is necessary for designing instructional strategies. Further, the subject domain to be trained, available technology, and basic research in related fields also

influence the selection of research topics and issues. This model has guided the development of the Automated Instructional System Research program.

Additionally, ATMRU continues to conduct workshops that look at key technical issues. It is currently organizing a workshop on behalf of NATO. This future workshop will focus on the use of advanced technology in synthetic training environments.

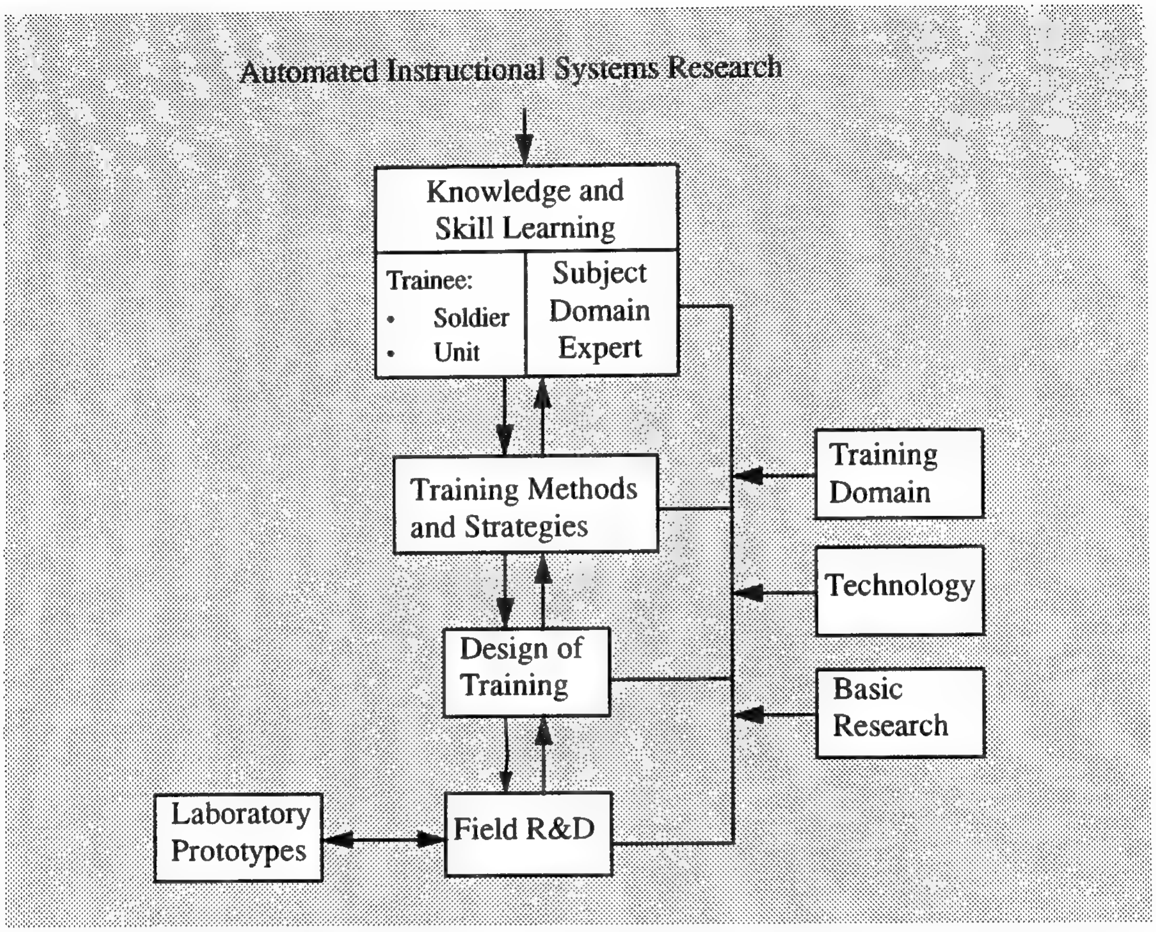


Figure 21. ARI Conceptual R&D Framework

6.4 On-Going ITS-Related Tasks

Two projects are discussed here. The first project does not explicitly address ITS development but is intended to develop knowledge that will guide the development of VR-based ITS. The second project address ITS support for second language learning.

6.4.1 Visual Knowledge Representation in Multi-Media CyberSpaces Project

VR offers many new opportunities for the role of an intelligent tutor. As stated in [Psotka 94]: "As a ghost presence, the tutor can interact with a student through digital speech, through text that floats in air, or through replays. As an embodied presence, the tutor can vary in reality from a stick figure to a realistic manikin, with facial expressions and voice. The possibilities for realistic guidance that is believable and as forceful as a real tutor's may be quite difficult to achieve, but dramatic in their implications. The possibilities envisioned for ITS are both made more concrete and expanded by the potential of VR."

The phenomenon of immersion, which is central to the perceptual experience of VR is poorly understood. Much needs to be learned about the educationally leveraging components, process, and effects of immersion. This project will investigate how emerging VR technology can be used as a tool for studying the psychological process of immersion and for applying the results to the development of synthetic training environments.

Better understanding of how instruction can take advantage of the multi-sensory input will benefit many future ITS developments. In the immediate future, the results of this research will be applied in the development of the Virtual Physics Tutor (Section 14.4.7 on page 190).

Programmatic Background

This project began in Fall 1992 and is due to complete in Summer 1994. The total funding is approximately $500K. ARI researchers are supported by researchers from the National Institute for Standards and Technology (NIST), Catholic University, George Mason University, George Washington University, and Georgetown University. The Principal Investigator is Dr. Joe Psotka from ARI.

Planned Products

Authoring tools for the construction of virtual environments (to be used as research tools) on Silicon Graphics and IBM PC machines.

Approach

In support of the overall hypothesis that *virtual environments turn knowledge into experience*, this basic research is organized around answering two main research issues:

- Identifying the individual differences that relate to how well people become immersed.
- Determining the benefits of VE on memory and comprehension for training.

It requires developing support for the construction of virtual environments as research tools. It also requires conducting experiments to determine (1) cognitive components of visual immersion, and (2) principles of training design in virtual environments.

The work is divided into three tasks.

Task 1: Develop paradigms for assessing fidelity of VE. **The relationship between field of view and "VR Egocenter" has been determined. The cognitive components of visual immersion have been identified.**

Task 2: Create paper and pencil assessments tests. **Two questionnaires have been developed to gather immersion-related information: one intended to provide insight into the process of immersion, and the other into the nature of "presence." A sample from one of these questionnaires is provided in Figure 22 on page 91.**

Task 3: Develop paradigms for assessing memory and comprehension. **Experiments on memory have begun.**

Potential Follow-On This research may have potential applications in Distributed Interactive Simulation. The next step may be a 6.2 effort to exploit the knowledge gained in prototype ITS systems.

11. How disoriented did you feel after the experience?

Totally	Very	Somewhat	A Little	Not At All

12. How disturbing was the lag or delay between your movements in the real world and the VR world?

Totally	Very	Somewhat	A Little	Not At All

13. How often did you feel your body image was in the wrong place in the VR world?

Totally	Very	Somewhat	A Little	Not At All

14. How responsive was the environment to your movements?

Totally	Very	Somewhat	A Little	Not At All

15. How natural and realistic was any object motion?

Totally	Very	Somewhat	A Little	Not At All

16. How much narrower was the field of view than normally?

Totally	Very	Somewhat	A Little	Not At All

17. How completely could you survey or search the environment visually?

Totally	Very	Somewhat	A Little	Not At All

Figure 22. Excerpt from Immersion Presence Questionnaire

6.4.2 Multilingual: Advanced Technologies for Mastering Foreign Languages Project

The military have many needs for mastery of foreign languages. For example, military intelligence linguists must use foreign language to collect intelligence critical to Army missions, multinational engagements (e.g., Desert Storm) require rapid cultural and language train-ups, and sudden international actions and interoperability increase requirements for foreign language skills. The goal of this project is to conduct R&D that will

enhance foreign language training through advanced instructional technologies and extend specific understanding of foreign language skill acquisition. Specific objectives include the following:

- Improve teaching of languages from skill levels 1 to 2.
- Teach In-MOS (Military Occupational Specialty) language skills not now taught for MOSs, for example, interrogation.
- Improve retention of MOS language skills.
- Increase opportunities to maintain language on the job.

Attainment of these objectives will save instructor time needed to teach and maintain MOS-specific foreign languages skills, increase language proficiency of Military Intelligence (MI) linguists, and increase their readiness. It will also improve foreign language applications in future systems such as portable job aids, machine translation systems, and interoperable communication systems.

Programmatic Background

This effort started in 1989 and will end in Fall 1995. The total funding is approximately $2.5M. Much of the early work was performed under a contract with SAIC. Much of the current work is being performed under a contract, MDA 9030920C-0229, with Micro Analysis & Design, Inc. The Principal Investigator for Micro Analysis & Design is Ms. Susan Dahl.

Participating organizations include US Army Special Warfare Center School, and the Defense Language Institute, Foreign Language Center. Additionally, ARI has entered a Memorandum of Agreement, entitled "Coordination of Efforts in R&D of Language Sustainment Training in Support of MI Courses," with US Army Intelligence Center and School, Ft. Huachuca (Department of Human Intelligence). The Deputy Chief and Staff for Intelligence (DCSINT) at the Pentagon is another participant.

Planned Products

Concept demonstration German Tutor, prototype Bridge tutor with dynamic interactive graphics. Research results yet to be determined.

Approach

This effort is investigating how advanced computer technology can be used for enhancing the functional capability and instructional effect of computer-based instructional systems in teaching a second language. It addresses theoretical and technical issues arising in the development of intelligent computer-assisted second language learning (ICALL), a new field that seeks to merge development in ITSs with advances in natural language processing (NLP). The overall approach has been to develop a demonstration product, the German Tutor (also called BRIDGE), to demonstrate basic concepts of designing and developing an authorable computer-based language tutor. A second tutor then builds on the earlier work to provide a prototype foreign language tutor, the Military Language Tutor (MLT), that can improve the training of military linguists.

Once developed, these tutors will be used in a research effort to collect data to examine the process of language learning and the effectiveness of different instructional strategies. Plans for this research have yet to be developed. Consequently, the following tasks focus on the development of MLT.

Task 1: *Review state of the art.* Conduct a conference to survey the state of the art of computer-based language training, and the government requirements for such training. **This task has been completed.**

Task 2: *Prepare a detailed and understandable design and complete screen-state prototype of the MOS (job)-specific language tutor.* This initial design will be independent of the target foreign languages, will be reconfigurable and authorable as feasible by non-programmers, will emphasize immersion-like training, and will assume a model of language learning that is explicitly justified in theories of learning and cognition, in relevant empirical research, and in current methods of language pedagogy. A screen-state prototype means that all classes of screens to be seen by students or used for authoring will be worked out in detail and presented to ARI using a prototyping software package. The projected completeness and ease of control by non-programmer, non-computational linguists will be a significant issues in the design. **This task has been completed.**

Task 3: *Prepare a detailed plan for integrating NLP technologies with the design from Task 2 for MOS-specific language training.* This plan will address a fully developed, language-specific instantiation of the tutor. The plan should be justified in detail by existing empirical research as well as linguistic and cognitive theory. The plan should include the following:

- Alterations, additions to, or replacements of existing language-specific software that are required for fully developed foreign language instantiations.
- The instructional goals to be dealt with.
- The classes of errors to be collected and how they will be detected or inferred.
- The mechanism and resulting screen-states that will permit a non-programmer who is not a computational linguist to enter, delete, and alter lexical entries.
- The initial tutoring rules and values to be used.
- The initial approaches to feedback and explanation and the controlling rules.

This task has been completed.

Task 4: *Implement the approved tutor design and integration plan for Arabic and Spanish.* **This task is in progress.**

Task 5: *Alpha test the implemented tutor and repair bugs.* Contractor and ARI personnel will engage in alpha testing. Bugs, structural anomalies, significant design flaws, and data errors will be fixed.

Task 6: *Beta test the prototype tutor at ARI and selected Army sites.* Beta testing will be conducted by a reasonable sample of training personnel and students. Their views will be captured. Bugs, structural anomalies, significant design flaws, and data errors will be fixed.

Task 7: *Plan and conduct research to improve the tutor design and to test the theoretical framework and the cognitive model of language acquisition underlying the design.* Plans for conducting the research will be developed in concert with ARI. These plans will include required facilities, sources of subjects, and a selection of significant and testable hypotheses.

Accomplishments For the project as a whole, the following has been accomplished to date:

- Development of HyperLexic, an in-house military vocabulary hierarchical tutor, first quarter FY91.
- Development of a German parser to support Intelligent Language Tutor, first quarter FY91.
- Development of a prototype Intelligent German Tutor for 97E M1, first quarter FY92.
- User trials of German Tutor at USAICS, second quarter FY92.
- Extension of parser to Arabic, third quarter FY92.
- User trials of German Tutor at Fort Bragg in the second quarter of 1993.

6.4.3 Military Language Tutor

The Military Language Tutor (MLT) is intended for use by military intelligence (MI) linguistics and special forces. Thus the content focuses on command, control, communications, and intelligence (C3I) job-specific language skills. The language focus is on two priority languages: Arabic and Spanish, with skills coverage including reading, writing, listening, and speaking, with treatment of speaking limited by available speech recognition software. The coverage of lessonware includes selected grammatical and other linguistic principles that characterize shortcomings on Level 1 linguistics and that are crucial in the transition from Levels 1 to 2. These principles are important to job performance. For example, a language training needs analysis conducted by ARI with regard to military interrogation students (MOS 97E) assigned priorities to a range of linguistic constructions, such as spatial prepositions, temporal adverbs, clause subordination, and politeness markers, based on their saliency in interrogations and their association with errors of students. The specific lessons types supported are:

- Multiple choice.
- Fill in the blank.
- Question and answer.
- Identify (multiple) location(s).
- Sorting.

- Menu-created sentences.
- Constrained single sentence entry with graphics.
- Translation (English to Arabic/Spanish and Arabic/Spanish to Arabic/Spanish).
- Dialog.
- Pronunciation practice.
- Speech recognition.
- Parser free play

A sample MLT screen is shown in Figure 23 on page 96.

To facilitate its use as a research tool, the tutor is designed to enable easy re-configuration and authoring (by a non-programmer) in several respects. These include dimensions pertinent for tutoring, for example, the nature of remediation methods. It also includes support for re-configuring and authoring of the lexicon.

The specific advanced technologies applied in the tutor include NLP, integrated with ITS and with such techniques as dynamic graphics, an interactive dialog that can support transactions simulating second language immersion. The NLP coverage includes a wide range of syntactic constructions as well as domain-specific aspects of semantics and pragmatics as needed to support a limited dialog.

Development Status

Prototype system under development.

Architecture

An overview of the MLT system is shown in Figure 24 on page 97. The following paragraphs discuss the most important system components and features from the perspective of the support they provide to the user of the system, in this case a lesson author, or instructor.

The MLT is a PC-based, MicroSoft Windows compatible product. When an author first enters the software, they are presented with a title screen. From this screen, they have the option to either start a lesson definition from scratch, or to resume work on an existing lesson. If they choose to start a lesson from scratch, they will be presented with an Index of Lessons.

Language-Based Index of Lessons

This component of the system is based on a data base stored set of lessons. the lessons are categorized by language (for example, Spanish, Arabic) and include a summary statement that the lesson author has composed. The user will be able to scroll through this index to identify the lesson they want to use. Existing lessons can be modified or they can be used as a basis for new lessons.

Lessons are composed of exercises. The MLT supports 14 different exercises types, as discussed previously. The exercises are linked together by the author

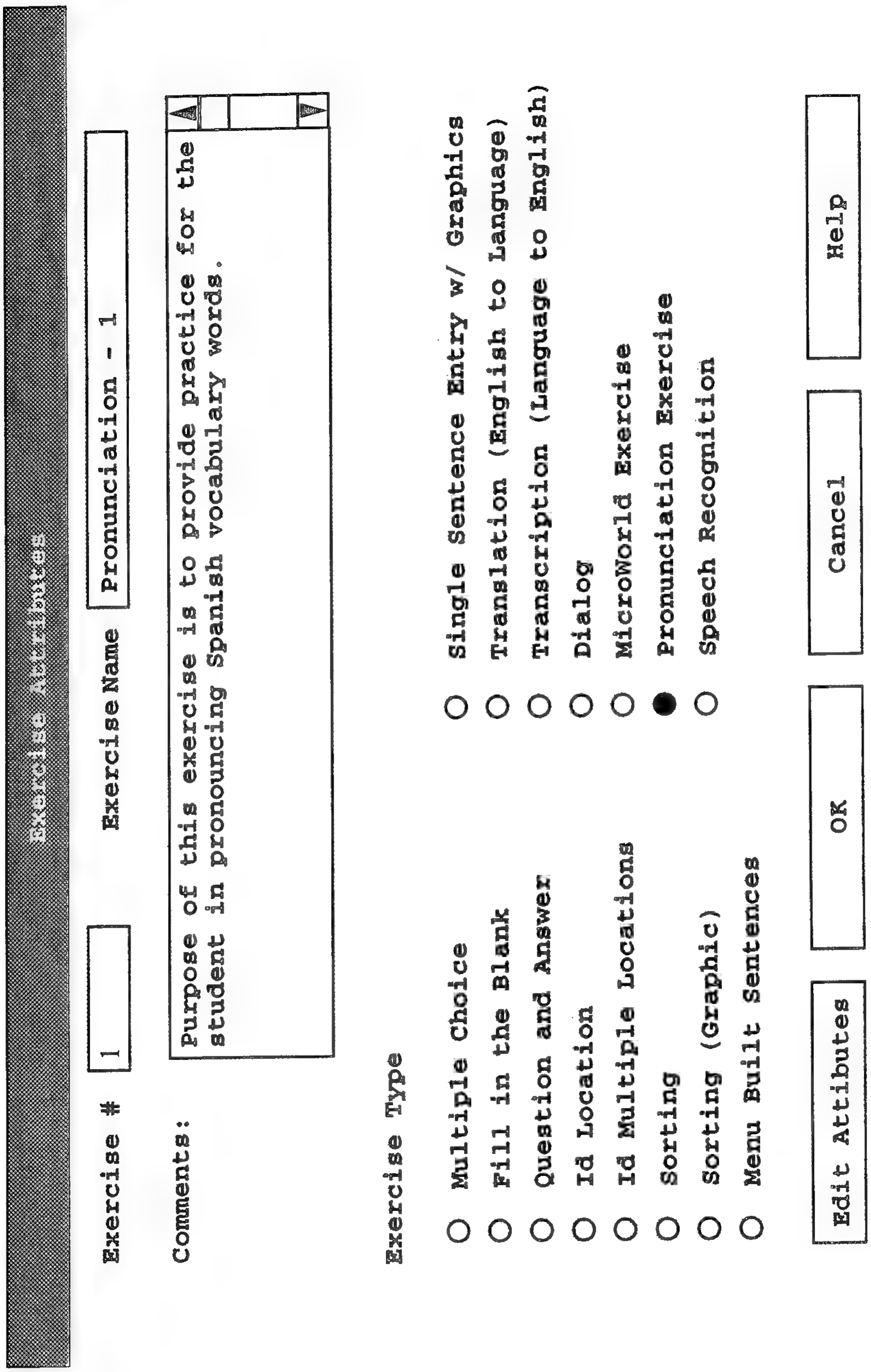


Figure 23. MLT Sample Screen

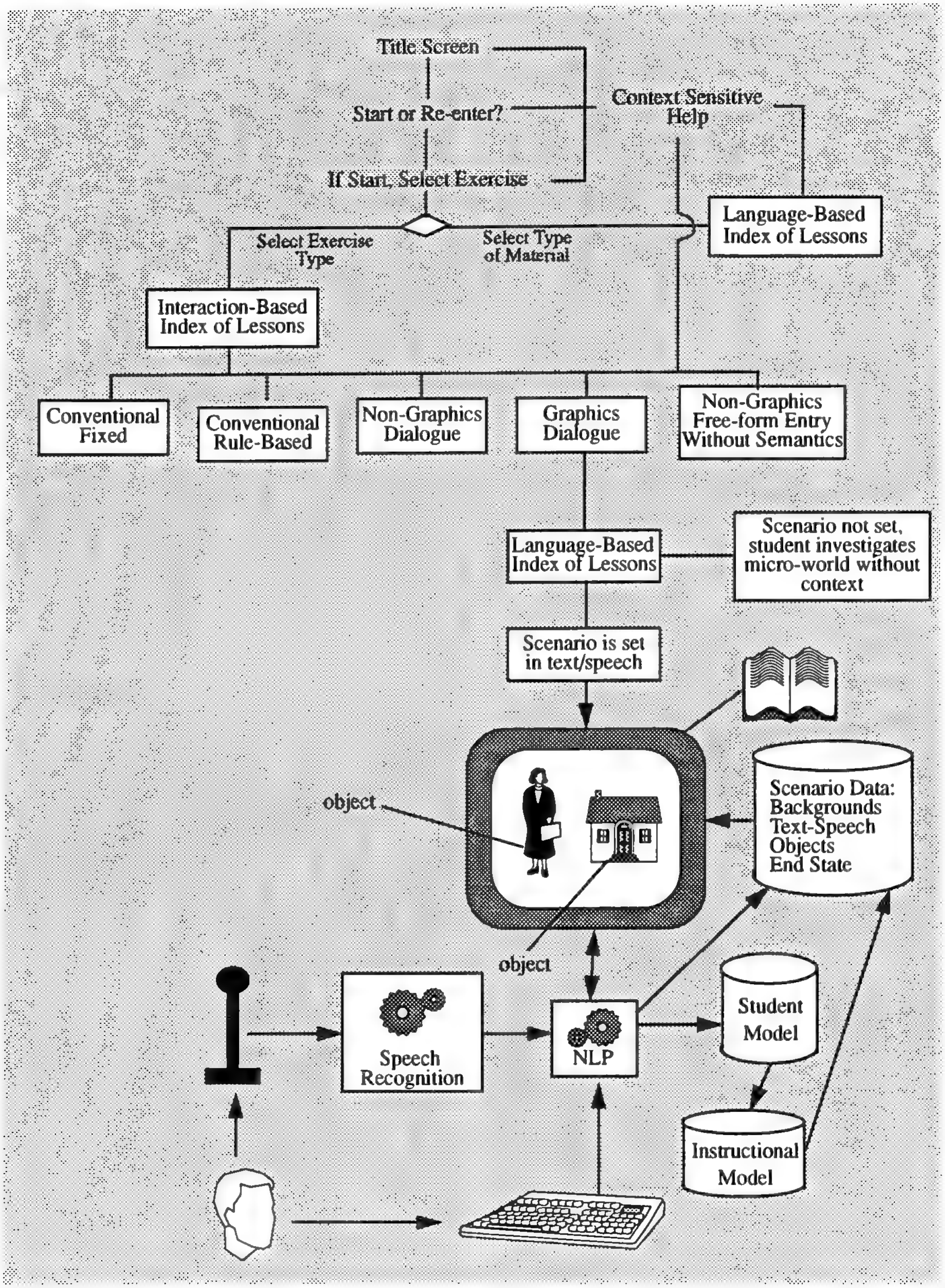


Figure 24. MLT Overview

using a variety of methods, ranging from conventional fixed to performance-based sequences.

Scenario Data Each lesson is supported by a set of scenario data. The scenario data support the individual exercises within the lessons. Scenario data consist of speech files, graphical backgrounds, graphical objects within the backgrounds, as well as maps and textual information. Scenario data also include scoring information that specifies the acceptable performance levels for a student to have successfully completed the lesson. Lesson authors can use scenario data that are packaged with the tutor, or they can develop their own.

Objects Objects are graphical elements that are initialized by the scenario data set and manipulated through the exercises by the student. Examples of objects are a person and a building. They might be used in an exercises where the student is instructed to type instructions in Spanish that would make the person enter the building. As the student enters the instructions, the MLT will move the objects in response. Feedback to the student is provided immediately and graphically.

Speech Recognition

As a minimum, the MLT will include single-word speech recognition. The author will be able to design exercises that will test the student's ability to speak in recognizable Spanish and Arabic accents. The future goal for this system is to include speaker-independent continuous speech recognition. When complete, this capability can support any and all exercises types.

Natural Language Processor Many of the exercises in the MLT are based on NLP functions. This component is the most advanced and technically challenging of the system. In the simplest terms, the NLP consists of a parser (to determine whether the student has entered a response that is grammatically correct), a semantics module (to determine whether the student's response is meaningful and accurate), and a discourse planner (to compose an appropriate response to the student). All three of these elements are required for dialogue exercises, but may not all be required for simpler exercises (for example, multiple choice).

Student Model The MLT does include sophisticated capabilities for modeling an individual student's progress through a lesson, or set of lessons. The author can compose branching logic to control the amount of remediation and the level at which remediation is required. Additionally, each student's records are maintained to a level that supports individual instruction and testing.

Instructional Model The MLT will be delivered with embedded basic instructional techniques and examples. They will provide an author with guidance on structuring remedial lessons, and for providing feedback to students at various levels. The tutor will also include sample lessons in each language.

Context Sensitive Help Both the student and the author will have constant access to a hypertext-like context sensitive help system. This system will assist users of both types on understanding how the system works and how to navigate through the MLT. The Help System will include an index, glossary, and graphical support for the standard help screens.

Operating Environment

IBM-compatible PC, 486-based, with Windows.

Evaluation Status

Plans have been made for alpha and beta testing to be performed once MLT development is complete.

Future System Development Plans

Future plans include the following:

- Integrating the German Tutor with MLT.
- Expanding MLT to provide instruction in additional languages yet to be determined.
- Moving beyond discrete speech recognition to support continuous language recognition, also speech synthesis.

7. ARMY RESEARCH INSTITUTE, AVIATION R&D ACTIVITY

7.1 Mission and Role of ARIARDA

The Army Research Institute, Aviation R&D Activity (ARIARDA) at Fort Rucker, Alabama, is a field operating activity of the Army Research Institute for the Behavioral and Social Sciences headquartered in Alexandria, Virginia.

The ARIARDA conducts aircrew training research and provides technical support to the Army Aviation Center in the area of aircrew performance and training. Current research is in five general areas: flight simulation, aviator selection and assignment, institutional and combat unit training, crew coordination training safety, and team training for emergency rooms. ARIARDA research efforts span the phases of the aviator's career cycle, commencing with aviator selection and initial entry flight training to Army-wide aircrew sustainment training.

This is the first ITS-related effort being undertaken. There is no overall ITS R&D program plan.

7.2 On-Going ITS-Related Tasks

7.2.1 Intelligent Flight Trainer for Initial Entry Rotary Wing Training Project

This SBIR Phase II effort is formalizing and expanding upon several of the concepts demonstrated in the Automated Hover Trainer program, with the development of the Intelligent Flight Trainer (IFT). Specifically, the IFT will integrate four key technologies: (1) ITS methods, (2) adaptive training techniques, (3) speech synthesis, and (4) low-cost simulation technology. The IFT will focus on primary phase Initial Entry Rotary Wing (IERW) training, with Transfer of Training (TOT) effectiveness measures providing the measure of overall system effectiveness in training.

The anticipated results of the Phase II effort are as follows:

- Development of a stand-alone IFT module for use with a full range of IERW maneuvers, and an interface specification protocol for integration with existing flight simulator trainers.

- Validation of system operation in existing research simulators. This will support an evaluation of IFT TOT effectiveness, and IFT operability in an advanced simulation environment.
- Generation of product prototype hardware and software specifications to transition the research prototype to development.

This effort will include defining the requirements for follow-on development and identifying options for scope expansion beyond IERW training.

Background

Charles River Analytics, Inc. has been awarded DOD Contract MDA903-C-93-0132 to perform this work. This SBIR Phase II contract runs from July 1993 to June 1995 and is funded for $500K.

Charles River Analytics is supported by subcontracts with engineers at the University of Alabama at Tuscaloosa. The Principal Investigator is Dr. Greg Zacharias from Charles River Analytics.

Planned Products

Prototype full-scope IFT.

Prior Work

The IFT effort builds upon previous R&D efforts to develop and fabricate a low-cost visual helicopter training simulator and to augment the low-cost simulator as an Automated Hover Trainer (AHT). The low-cost training simulator, called the UH-1 Training Research Simulator (UH-1TRS), was developed and fabricated by the University of Alabama at Tuscaloosa under contract to ARIARDA. The UH-1TRS has been evaluated as an *ab initio* trainer in several experiments. The data demonstrated that the UH-1TRS produced positive TOT to the UH-1 aircraft in randomly sampled Army IERW trainees. Experimentation also showed that the UH-1TRS could substitute for actual aircraft time in IERW Primary Phase training.

The first efforts in the direction of an intelligent trainer were realized with the development of the AHT in FY91-92. The AHT used adaptive training to instruct *ab initio* trainees in the five basic hovering maneuvers: stationary hover, hover taxi, hovering turns, land from hover, and takeoff to hover. The AHT synthesized an augmented helicopter aerodynamic model through the application of Optimal Control Theory to a feedback loop that rendered an "easy to fly" helicopter model. As student pilots progressed in the training and exhibited less tendency to overcontrol the trainer, the automation software reduced the amount of control augmentation until student pilots were able to control the entire UH-1 aerodynamic model. The AHT demonstrated significant positive TOT to hovering flight in the UH-1 aircraft, with Army student pilots serving as research subjects.

The success of the low-cost trainer and the AHT application provided the impetus to develop and evaluate the IFT. The IFT was conceptualized by ARIARDA and the concept was brought to fruition by Charles River Analytics under an SBIR Phase I contract. A demonstration of the IFT concept showed the potential of the

concept and provided justification for undertaking a Phase II effort to construct and evaluate a full-fledged IFT device.

Approach This effort consists of four tasks.

Task 1: *Expansion of IFT Scope.* The researchers will expand the scope of the IFT developed under the Phase I effort via three subtasks: expansion of the scope to include the full range of IERW maneuvers, enhancement of component system capabilities, and improvement of software environment and hardware interface to the host simulator. The first subtask will broaden the range of IERW maneuvers and will include additional knowledge engineering with instructor pilots and subsequent codification of maneuver definitions, criteria, and diagnostics in the IFT rulebase. The second subtask will enhance the IFT capabilities to include a broader scope of evaluation, help, and advisory functions supplied by the IFT within the hybrid algorithmic/neural net/expert system environment. The third subtask will update the system software using a unified development environment. This subtask will also simplify the hardware interfacing requirements by migrating all of the IFT software to a single PC host. Testing and evaluation of this expanded scope IFT will make use of the UH-1 TRS simulation facility and tests will be made to ensure code validity and simulation compatibility. **This task is in progress.**

Task2: *Validation of Training Effectiveness.* The researchers will validate the enhanced IFT via a training effectiveness evaluation using two simulation subtasks. The first subtask will evaluate IFT performance in the UH-1 TRS environment, in terms of the system's capability to minimize student training time in the different maneuvers. They will use iterative design and tuning of the adaptive training strategy and advisory information presented to the student. The second subtask will evaluate IFT training effectiveness, using a TOT paradigm and in-flight post-IFT evaluation. Training effectiveness ratios (TERs) will be used to evaluate IFT effectiveness. Direct comparisons with previous TER studies with the UH-1TRS will be used to evaluate the IFT's contribution to system training effectiveness. **This task is in progress.**

Task3: *Demonstration of Operation with Advanced Simulator.* The researchers will demonstrate IFT operation with an advanced rotorcraft simulation at Fort Rucker. They will demonstrate the process of interfacing the IFT module with the existing flight simulator and demonstrate system operation. They will modify the IFT helper module to provide augmentation appropriate to the simulated vehicle, patch the vehicle model to support the IFT monitoring and augmentation functions, and install the necessary communications interface. Following this system update, a demonstration of system operation will be conducted to evaluate training effectiveness in the advanced simulation environment. If warranted. a restricted scope follow-up TOT experiment will be conducted to demonstrate transfer effectiveness of the upgraded system. **This task is in progress.**

Task 4: *Requirements Specification for Full-Scope IFT.* The researchers will generate requirements specifications for a full-scope IFT for general use by the rotorcraft flight simulation training community. This task will focus on the development of functional specifications based on the research-oriented software developed under

Phase II. The requirements specifications will be for an integrated, modular, and expandable package that can be readily customized for the specific host simulator. Further development directions will be identified for a Phase III effort aimed at production development of a modular IFT system for generic interfacing with existing simulators.

Potential Phase III Follow-On

The Phase II development, validation, and product specification tasks will provide the foundation for Phase III development of a knowledge-based Intelligent Flight Trainer module targeted for integration with a number of existing rotorcraft flight simulators. The hardware and software specifications generated under Phase II would be implemented under Phase III, leading to a stand-alone IFT module and interfacing protocol for upgrading existing systems. Demonstration and evaluation of this package would be guided by the evaluation methodology employed under Phase II, so that a critical early evaluation of the Phase II prototype could be made before significant development effort is expended. The prototype would then proceed to full product development, targeted at one or a few high-volume simulation facilities. In the Phase III effort, the researchers would also plan to explore options for direct integration of the IFT with future training simulators during the simulator development process.

The researchers see considerable potential in the commercial area for a wide range of vehicle training simulators. The basic technology could be directly applied to initial entry training in private fixed-wing and rotary-wing training programs, with considerable savings to be expected in facilities expenses (via low-cost technology) and instructor personnel expenses (via automation). Enhancement of the knowledge base of the IFT would allow it to be used in commercial transition training as well, especially in the practice of unusual or emergency maneuvers. A scope expansion would also support annual proficiency training, as a part-task adjunct to IFT evaluators.

A considerably greater commercial potential exists, however, for a low-cost automated automobile driving simulator/trainer. The market is large, the potential cost savings in capital and labor large, and the technology can support extensive defensive driving training in scenarios that would never be attempted in the real-world environment used on a conventional driving syllabus. Further, the market is not limited to initial driver training; if used in a refresher program or as part of a license renewal screening program, it seems clear that substantial savings in lives and property costs could be accrued yearly.

The researchers see a number of potential Federal Government applications of the proposed effort. Their concentration in this work has been aimed at developing a low-cost visual flight simulator for primary phase IERW training, but clearly a similar approach could be used for development of rotorcraft simulators used by the other services, in particular the Navy. This approach would be especially cost-effective if done in conjunction with rotorcraft simulation and training development efforts now underway at the Naval Air Warfare Center Training Systems Division (NAWCTSD), particularly in the application of AI technologies to ITS.

There are also potential military applications in a number of non-rotary-wing areas. The fixed-wing area is particularly appropriate, since a tactical IFT would allow for self-paced training on a number of critical offensive and defensive tactical air superiority maneuvers, and could support a wide range of knowledge bases on how best to accomplish these maneuvers successfully, as expressed by a number of tactical domain experts. A low-cost stand-alone trainer of this type could be used for proficiency training on-base, or even closer to the tactical scenario. It would be available 24 hours a day, be completely self-paced, and encapsulate the knowledge of many tactical experts and effective instructor pilots.

7.2.2 Intelligent Flight Trainer

The student pilot operates in the closed-loop flight task, generating flight controls in response to a multi-modality cueing system, which can provide visual, motion, proprioceptive, tactile, and auditory cues. The student's controls drive the helper portion of the IFT which augments the student controls to generate an *aided control set*. This aided control set, in turn, drives the vehicle simulation proper. The resulting vehicle state is fed back to the cueing system for conventional closed-loop flight control operation by the student pilot.

The vehicle state is used for computation of the student's performance levels. The IFT then relates student performance to desired maneuver criteria and, on the basis of a computational pilot model, generates an appropriate help level to be provided [Krishnakumar 1991]. Auditory messages are generated and sent to the student pilot through the cueing system, on the basis of a prestored message database, and in response to a rulebase regarding particular advice for the given vehicle states and student performance levels. The rulebase in turn is generated from a knowledge engineering effort based on criteria identified by the instructor pilot. A GUI provides for an interface between the IFT executive and the human IFT supervisor.

Figure 25 on page 106 and Figure 26 on page 106 illustrate how the IFT's adaptive autohelper varies the difficulty of the hover task as a function of changing student pilot proficiency over time. Figure 25 shows the longitudinal performance index evolving over time; large values indicate poor performance. For this series of runs, the student is starting with no help whatsoever, (help level 0) and the help level is gradually stepped up by the IFT as the evaluator recognizes the student pilot's need for assistance in flying the vehicle. For the first 20 seconds, the student is on the ground. Auto takeoff occurs just before 20 seconds into the simulation, and the student immediately incurs large longitudinal errors. The system gradually increases the help level from 0 to 1 to 2 at approximately 30 seconds into the run. This results in an improvement in student performance, but the flight still exceeds the

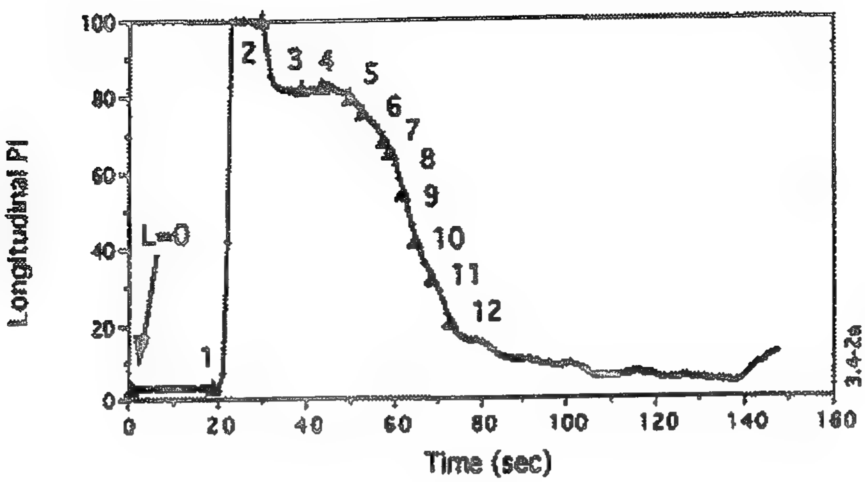


Figure 25. Longitudinal Performance Index Over Time

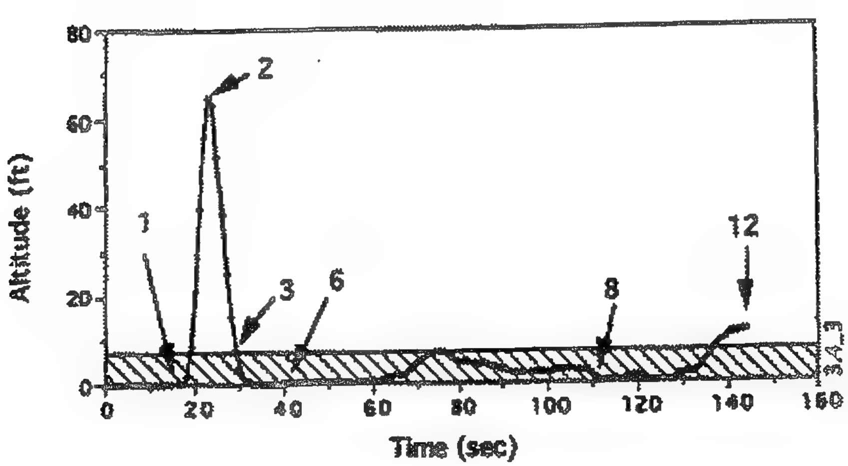


Figure 26. Altitude Profile Over Time

criterion level, so the helper continues increasing the help level. At approximately 50 seconds, the student begins a gradual and continuing improvement in performance; but the helper continues increasing the augmentation level, so that by 70 seconds into the run, the augmentation level is at its maximum level of 12. At this point, the student continues improving flight control so that at the end of this demonstration (140 seconds), the student appears to be well on his way to bringing the vehicle into criterion level on the longitudinal axis.

Figure 27 on page 109 shows the altitude profile for the same training run. Table 3 on page 108 shows the corresponding message sequence generated by the advisor for the altitude profile. Following takeoff at approximately 17 seconds into the run, the student quickly incurs a high climb rate. At approximately 19 seconds, he exceeds the upper bound on altitude, and the advisor generates a message informing him that he is too high. A few seconds later, as he peaks out at over 60 feet altitude, the advisor advises him to descend using down collective. Later as he is approaching the criterion bounds, the advisor again tells him he is still high. In the 30 to 60 second time interval, the student is still below the desired criterion altitude, and a series of messages are generated. After this series of advisory messages, at about 65 seconds into the run, the student maintains altitude for almost a full minute within the criterion limits. At about 110 seconds, he begins to exceed the lower criterion limits, and a second series of messages is generated.

Development Status

Prototype was completed in Phase I of this effort. Fully functional system under development.

Architecture

ITS commonly structure their knowledge into three data base [Steinberg 1991]: (1) the instructional or teacher model, (2) the subject matter or domain expert model, and (3) the student model. To take advantage of this structuring in the flight simulation context, the researchers in Phase I showed how the IFT components could be organized along these lines. Table 4 on page 108 illustrates the mapping.

Figure 27 on page 109 illustrates the architecture of the IFT components. Four basic levels of IFT are shown.

Top Level

The top level represents the basic flight simulator, whose two major components are the student pilot and the vehicle simulation itself. The figure shows an interaction from the student to the simulation via the student's controls, and from the simulation back to the student via the multi-modality cue provided by the simulator cueing system.

Second Level

The second level in the IFT architecture represents the basic instructional/teacher model. Here three components are shown. First the evaluator/helper monitors the vehicle state generated by the vehicle simulation, and on the basis of an evaluation of student performance, generates an assist level sent back to the vehicle simulation. As described below, the purpose of this level is to adaptively modify the vehi-

Table 3. Message Sequence Generated by IFT

TIME (sec)	MESSAGE SEQUENCE	MESSAGE
19	1	"You're too high; descend a bit"
24	2	"Descend a bit, using down collective"
30	3	"Check your altitude, match the truck window height"
32	4	"Check your altitude"
38	5	"You're too low"
42	6	"You're too low, climb a bit"
50	7	"Climb a bit, using up collective"
*		*
*		*
*		*
112	8	"Check your altitude, match the truck window height"
116	9	"Check your altitude"
122	10	"You're too low"
130	11	"You're too low, climb a bit"
140	12	"Check your altitude"
*		*
*		*
*		*

Table 4. ITS Knowledge Bases and IFT Components

ITS Knowledge Base	IFT Components
Instructional/Teacher Model	IFT Executive Strategy: Provides for overall control of the IFT. Performance Evaluator: Monitors student performance during the specific maneuvers being taught. Student Helper: Assists the student in accomplishing the maneuvers. Student Advisor: Provides advice across a range of perceptual and motor strategies.
Domain Expert Model	Vehicle Knowledge: Provide in-depth knowledge of the vehicle dynamics that the student is trying to control. Maneuver Criteria (Expert Model): Used for judging flight control "goodness" at the levels that might be accomplished by a proficient or expert pilot.
Student Model	Student Pilot Model: Provides for a representation of on-going improvement in student proficiency as learning progresses.

cle characteristics so as to gradually increase task difficulty as student proficiency improves (and conversely reduce task difficulty as proficiency slips). Student performance generated by the evaluator is sent to the second component, the advisor, which uses an advisory ruleset to generate spoken advice for the student pilot. This advice is delivered via a synthetic voice system. The third component shown is the IFT executive, which controls both the advisor and the evaluator/helper, to effect synergistic advice and adaptive training.

Third Level The third level shown in the architecture composes the IFT knowledge base. It consists of the maneuver criteria and student pilot model used by the evaluator/helper to both judge student performance relative to expert or criterion level performance, and to infer student pilot proficiency on the basis of observed performance and known pilot capabilities. Also shown at this level is a message database feeding the advisor messages for subsequent transmittal to the student pilot.

Fourth Level The fourth level of the architecture shows a graphical user interface between the IFT executive and the instructor supervisor. The intent here is to provide for supervisory control of the IFT, and for modification of the IFT strategy and/or knowledge base.

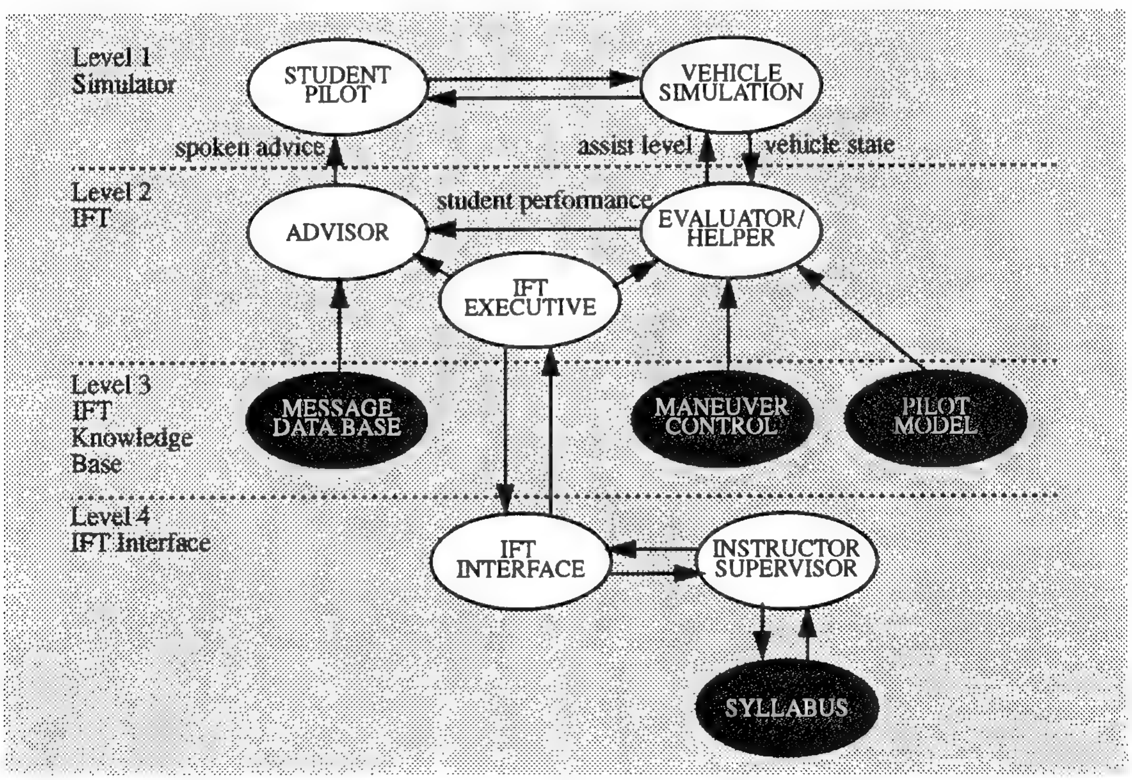


Figure 27. IFT Architecture

Operating Environment PC-based environment yet to be determined.

Evaluation Status

A limited number of evaluation studies were performed with a prototype version of the IFT in the Phase I part of this effort [Zacharias 1992]. The results collected in the course of these evaluations show the following:

- On-line performance evaluation by the IFT can provide a simple indication of student performance across multiple dimensions. Only two scalar parameters, one for longitudinal axis performance and one for lateral axis performance, need to be specified.
- Adaptive aiding by the helper portion of the IFT shows how levels can be automatically adjusted upward with student needs and adjusted downwards with student proficiency.
- The expert system emulation of instructor pilot advice and student pilot diagnosis provides the student with insightful suggestions as how to improve performance when the evaluation indicates large errors, and the helper has reached its maximum help level.
- The combination of adaptive aiding and intelligent tutoring act synergistically to improve student performance in a short period of time, so that students minimize their need for advisory messages and minimize their dependence on the helper for system augmentation.

Although these evaluations were conducted over short learning intervals and did not lead to full proficiency on the UH-1TRS, researchers anticipate that a full scope validation effort will show effective training under this paradigm, as has been demonstrated in the precursor studies.

8. DEFENSE MODELING AND SIMULATION OFFICE

8.1 Mission and Role of DMSO

The Defense Modeling and Simulation Office (DMSO) was established 1991. Its mission is to "Strengthen the use of modeling and simulation in joint education, training, and military operations; research and development; test and evaluation; analysis; and production and logistics." [DOD 1991] DMSO's overall goal is to enhance readiness and make better use of limited resources.

With respect to education and training, DMSO has a single R&D effort underway addressing the challenge of cost-effective transition of DOD information technology to the non-DOD education and training community. DMSO has sponsored no past work in this area. There is no overall R&D program for ITS-related work.

8.2 On-Going ITS-Related Tasks

8.2.1 Analytic Methodologies for Modeling and Simulation Project, Task 4: Building Education and Training Technology on Commercial Software

This task will investigate the feasibility and cost effectiveness of developing ESSCOTS (Educational Software Systems Built on Commercial-Off-the-Shelf Software) applications. It focuses on the practical side of the technology transfer issues implied by a broad attempt to move software developed for commercial and military markets into an educational market—a market that is fractured and generally unappealing to for-profit software developers. (See also Section 15.5.1 on page 219, which discusses a distinct but complementary project that addresses theoretical and evaluative ESSCOTS issues.)

An ESSCOTS application begins with an existing commercial (or military) system, then adds a software "wrapper" that turns the application into an education or training environment. Modeling and simulation (M&S) now used by analysts are powerful tools for making military decisions, yet the cost of training analysts to effectively use these tools is high. This task aims to develop an ESSCOTS based on such M&S systems that will reduce the cost of training analysts, increase the number of analysts able to effectively use the

M&S, and thus potentially improve the quality of decision-making. The proposed work will draw from two current RAND research strands. The first is RAND's simulation community, which continues to develop state-of-the-art modeling technology. The second is RAND's educational community, which is building ESSCOTS for public education. This research has recently developed and field tested an ESSCOTS built on ARC/INFO, a geographic information system; this ESSCOTS enables junior-high and high-school students to learn geography, demographics, and statistical skills well beyond their grade level. In particular, the current effort will focus on the education and training opportunities of COTS and Government-off-the-shelf (GOTS) tools (such as Janus, VIC, DIS models) for simulation and modeling that have been developed over the past few years. This effort will focus more on GOTS and technology transfer from military applications, and will take a broad look at what commercial and military systems could be used as a basis for ESSCOTS.

In general, ESSCOTS have many things to recommend them; they are cost effective to develop (since they do not have to be built from scratch but rely on existing software); they covert single-use technologies to dual or multiple uses; they create natural alliances between businesses that develop COTS and education and training institutions; they train learners with the technologies they will use in their jobs; and they may act as a forcing function to shape interoperability of software.

In spite of their promise, much needs to be done before ESSCOTS have a dramatic effect on education and training productivity. It is necessary to identify which COTS can serve as the basis for effective educational systems, identify specific educational outcomes they can affect, develop broad principles and identify standards, if possible, to guide the development of the educational wrappers, build prototype ESSCOTS to understand how much they cost (compared with alternative "from-scratch" systems), and test these prototypes to see how they improve education outcomes. Once these steps are accomplished, researchers will be in a position to develop and implement ESSCOTS on a broad scale.

This work promises to broaden the scope of previously conducted military work by looking outside of the Army to other Services. The study will also benefit from the synergy of bringing together existing strands of non-defense and military education and training research.

Programmatic Background

This effort is sponsored by DMSO and funded by Director, Defense Research and Engineering (DDR&E). It is part of DMSO Project Number 0180 that continues support in FY94 for an on-going effort to assist the DMSO in building and implementing its long-term program. Overall, the project will allow RAND/NDRI to directly address some critical M&S research issues as well as to continue to par-

ticipate in a variety of working groups. The tasks to be undertaken include work on the use of new technology for education and training relevant to the DOD mission. Task 4, Building Education and Training Technology on Commercial Software, the topic of this subsection, is allotted $400K of the total funding amount. Work on Task 4 began January 1994 and will end in December 1994. The Principal Investigator for this work is Dr. David McArthur from RAND.

Planned Products Prototype ESSCOTS.

Prior Work The idea for ESSCOTS was formulated in previous NSF-funded projects (MDR-9055573), although the term *ESSCOTS* was not introduced until later. This early work included the development of the first ESSCOTS, which was based on the geographical information system ARC/INFO.

Approach The following steps will be undertaken in FY94.

Step 1: *Hold a Conference That Includes Both Experts in COTS and GOTS (From Business and the Military) and Experts in Education and Training*. The goals of the conference will be to identify useful COTS/GOTS products for specific education and training needs, as well as to formulate general principles and identify standards, if possible, for developing the educational wrappers that make COTS/GOTS into ESSCOTS. Emphasis at the conference will be given to modeling and simulation COTS/GOTS that appear to have particularly promising applications. Various approaches to developing educational wrappers will also be emphasized, including the importance of graphical interfaces, visualization, and systems that reveal progressively more complexities to students as they advance. **A range of COTS that might be the basis for ESSCOTS have been examined. The development of criteria that will help decide what makes a COTS a good candidate for an ESSCOTS has begun. Plans for the conference on COTS/GOTS and ESSCOTS have been made.**

Step 2: *Write Report*. This report will summarize the conclusions of the conference, addressing the characteristics of COTS/GOTS that will make them useful in education, which education and training needs ESSCOTS are best suited for, how ESSCOTS can be developed from promising COTS/GOTS, and the various costs and benefits of this dual-use approach to education and training technology development.

Step 3: *Guided by Conference Results, Select a Promising COTS/GOTS for Simulation or Modeling and Develop a Prototype ESSCOTS Based on It.* Special emphasis will be placed on selecting a COTS/GOTS product on which it is difficult to train analysts to use, and one that is now being used for analysis at RAND. The prototype ESSCOTS will be formatively evaluated in a lab setting.

Although the demonstration ESSCOTS developed here will be widely available, the main results of the research will be a set of general principles that can guide the development of future ESSCOTS, their wrappers, and, to some extent, may influence design of the underlying modeling or simulation systems. The primary audience for these results will be analysts and decision-makers in each of the services, for example, CAA and TRADOC. The work should also be of interest to various training and research audiences, such as Army CTC, PM-TRADES, HQDA AI Center, and ARI.

Two COTS have been identified as prime candidates for use as the basis of an ESSCOTS: RDSS, a model of why groups revolt, and SIMHEALTH, a simulation for studying and designing alternative health systems.

9. OFFICE OF NAVAL RESEARCH

9.1 Mission and Role of the Science and Technology Directorate

Public Law 588 of the 79th Congress established the Office of Naval Research (ONR) in August 1946 "to plan, foster, and encourage scientific research in recognition of its paramount importance as related to the maintenance of future naval power, and the preservation of national security." The mission of ONR remains faithful to that Congressional tasking.

The ONR science and technology investment strategy places particular emphasis on investing in ocean sciences, advanced materials, and informational science, as well as many broad and flexible programs that target all areas of science with potential Navy or marine Corps relevance. This strategy allows ONR to create and respond to scientific breakthroughs, develop and evaluate promising technology—and provide the cutting edge systems, platforms, and skills that will serve the Fleet well into the 21st century. The ONR program represents a $1.5 billion annual investment in support of the future Navy, Marine Corps, and the nation.

ONR's science and technology program also emphasizes working with industry from the earliest stages of research to accelerate the transition of scientific discovery into follow-on technological development. This is done by investing in programs that satisfy the Navy and Marine Corps customer needs in a way that also facilitates the commercialization of the technology.

9.2 Summary of Past ITS Work

ONR has been a key supporter in the development of ITS technology. Indeed, ONR sponsorship of John Anderson's development of ACT, the most complete and adequate cognition theory yet developed, helped to prepare for the way for the initiation of ITS field. Since then, ONR has supported many of the key researchers and key research efforts in this area. These efforts have ranged from addressing basic technology issues, such as identification of misconceptions, to application issues, such as the effect of ITS upon the roles of students and teachers.

ONR-sponsored research efforts have resulted in such notable products as Anderson's Lisp Tutor, which was the first intelligent tutor to attain the status of practical instructional utility and which now has been in practical use for several years. This work included the first systematic field evaluation of an ITS, conducted at Carnegie-Mellon University in Fall 1984. ONR also supported the basic research performed by Dr. Alan Lesgold, at the Learning Research and Development Center of the University of Pittsburgh, including research associated with Sherlock (see Section 2.4.8). This particular research now is being exploited by the Air Force, which has decided to invest to build a set of tutors similar to Sherlock. Together with NPRDC, ONR supported the development of the Intelligent Maintenance Training System produced by the Behavioral Technology Laboratory of the University of Southern California under Dr. Douglas Towne. ONR also supported Dr. Govindaraj of the Georgia Institute of Technology in producing a tutor for the diagnosis of faults in ship steam power plants.

Current AI tutoring technology differs from the behavior of human tutors in one very striking feature: no artificial tutor is yet capable of full natural language interaction, whether written or spoken. This precludes such tutors being able to match the full effectiveness of human tutors. Natural language interaction capability and issues in instructional strategy are the current emphases in ONR basic research relevant to intelligent tutoring.

9.3 Overall ITS R&D Program

Work on ITS technology is supported through the ONR STD's Cognitive Science Program which includes both a Base Program in cognitive science and associated Accelerate Research Initiatives. The Cognitive Science Program is managed by the Personnel Optimization and Biomolecular Science and Technology Department. It aims to provide a theoretical understanding of the human learner and performer in the domain of complex cognitive skills. This general goal unfolds into several interrelated, more specific objectives:

- To provide a theory of the fundamental characteristics of the learner and performer as an information processing system, including a theory of the basis of individual differences in cognitive abilities.
- To provide a theory of the nature of acquired knowledge and skill involved in performing complex problem-solving and decision-making tasks.
- To provide a cognitive learning theory that can account for the way in which such complex, structured bodies of knowledge and skill are acquired.

- To provide a precise theory of instruction, founded on cognitive theory, to be used to guide effective education and training of complex cognitive skills such as those involved in performing Naval duties.
- To provide theoretical foundations for personnel testing and assessment.

Under STD's Knowledge Acquisition Accelerated Research Initiative, there has been a major emphasis on AI-based models of complex human learning. This is reflected in a new Accelerate Research Initiative called Hybrid Architectures for Complex Learning. Artificially intelligent, computer-assisted instructional systems as well as more conventional instructional settings are the application areas for the program. In addition, projects are supported which involve either fundamental advances in AI bases for intelligent tutoring or the use of ITS as a laboratory for investigation into general issues of learning and instruction.

9.4 On-Going ITS-Related Tasks

9.4.1 Computer Dialogue Generation to Support Multiple, Sophisticated Tutoring Tactics Project

The generation of natural language tutorial interaction is one of the most significant open problems in intelligent computer-assisted instruction. This project exploits a collaboration with expert tutors who provide examples of human-generated tutorial guidance in using an instructional simulation, a typical context for intelligent computer-assisted instruction. The major goal of this effort is to learn how to produce intelligent computer-generated tutorial dialogue that responds to the user's needs.

Researchers set out to capture human tutoring dialogues, analyze the tutoring language and tutoring strategies, and build an ITS for a physiology domain. This system, CircSim-Tutor, can carry out an interactive natural language dialogue in which the system explores students errors with them and discusses the issues involved in the physiology and the problem solving algorithm. The system also probes for misconceptions and attempts to remedy them.

The researchers propose to build a system that generates many different types of interaction, including questions, positive, negative, and neutral acknowledgments, hints, and responses to student initiatives. They also propose to provide multi-sentence discourse at several points in the tutoring: summaries at the end of complex arguments and explanations of ways in which cause and affect mechanisms work, and remediations of student mis-

conceptions. The system will generate multi-turn structures, particularly hints and directed lines of reasoning, in which the tutor asks a series of small questions to prompt the student. The remediation of misconceptions requires multiple explanations with questions and answers interspersed among them. They plan to extend Rhetorical Structure Theory to allow the generation of representations of content and intention in parallel.

The researchers have built a Prolog prototype and two complete versions of the system in Lisp. They are now building a third version of the system with improved language capabilities and a multi-level knowledge base, covering a curriculum that can be restructured in response to the user's performance. They are also exploring the use of novice tutors, peer tutors, and tutors from other areas of the medical curriculum, to try to understand how differences in language and experience affect the tutoring process. To generate more intelligent, less repetitive surface structures, they propose to make more effective use of Lexical Functional Grammar (LFG) to carry out syntactic and lexical choice at the same time. The researchers also propose further studies of tutoring language and behavior.

Programmatic Background

The work is being performed under grant N00014-89-J-1952, awarded to the Illinois Institute of Technology, with a subcontract to Rush Medical College. The work began in November 1993 and is due to complete in November 1996. It is a 6.1 effort, with a total funding amount of approximately $379K.

The Principal Investigator is Dr. Martha Evens from the Illinois Institute of Technology.

Planned Products

CircSim-Tutor Version 2 is functional; Version 3 is in progress. Forty-five keyboard-to-keyboard tutoring sessions, each one to two hours in length. The Computer Dialogue System for keyboard dialogue capture. Concept Map Builder. A-Base, a system for tutoring medical students or corpsmen in the physiology of Acid-Base problems.

Prior Work

The work described here began under an earlier contract entitled "Computer Generation of a Tutorial Dialogue" which was funded from June 1989 to May 1991 with a budget of approximately $234K. Two contract renewals extended the contract to November 1993, with an additional total budget of approximately $165K. In addition, the Illinois Institute of Technology has received Augmentation Awards for Science and Engineering Research Training (AASERT) funding for two graduate students beginning June 1992.

During this period, the researchers have built a series of ITS on the basis of their analysis of the tutoring language and the tutoring methodology revealed in human tutor sessions. These sessions were carried out by Joel Michael and Alan Rovick, Professors of Physiology at Rush Medical College, who have served as expert tutors and domain experts, as models for the tutoring system and as sources of knowledge about the scientific domain, about the teaching of causal reasoning, and about the tutoring process.

Approach Three principles have guided the approach to the construction of ITS: (1) basing every aspect of the system on human tutoring sessions; (2) evolving ITS from a computer-aided instruction system (CIRCSIM), rather than starting from scratch; and (3) continuous availability of a working system to try out with medical students at every opportunity. While it has not been possible to have a system available at all times, the evolving system has been tried out on medical students frequently.

Since many tasks are revisited in the development of different system versions, the following discusses tasks for the entire research effort.

Task 1: *Tutoring Sessions and Capture System.* The first step was to capture human tutoring sessions; some are face-to-face, most are keyboard-to-keyboard. **Both types of human tutoring sessions have been recorded and analyzed. In the absence of adequate software, a computer program was developed for managing and recording the keyboard-to-keyboard sessions. This capture system, the Computer Dialogue System, records both sides of a keyboard-to-keyboard session conducted by two people sitting at personal computers with Hayes modems over telephone lines or any other link [Li 1992]. Researchers are currently recording more sessions, using novice tutors and tutors in other medical areas outside physiology that also require causal reasoning.**

Task2: *Analysis of Human Tutoring Language.* This analysis is the basis of both the natural language understanding and the generation components of the tutoring system. **The analysis showed that tutoring discourse is vastly different from ordinary human conversation and is planned in large multi-turn chunks. Captured tutoring sessions have been analyzed for such diverse phenomena as spelling errors, the syntax of the tutor's and student's language, positive and negative acknowledgments, Socratic questions, hints, repair of student misconceptions, responses to student initiatives, and explanations of cause-and-effect processes.**

Task 3: *Instructional Planning.* An ITS that emulates expert human tutors requires an instructional planner that determines what problems to pose and what strategies and tactics to use. It also decides when to ask a question, when to give a hint, and when to give a detailed explanation. These plans change dynamically depending on the confusions and misconceptions revealed by the student. Whenever the student asks a question or displays some other initiative, the planner must shelve its current plans and develop a new response. **Version 2 has a sophisticated multi-level planner that constructs plans dynamically using hierarchically organized sets of rules. The planner in Version 3 will make dynamic choices of the curriculum for sessions that last two or more hours in length, using information captured by the tutor.**

Task 4: *Student Modeling.* The computer tutor maintains a dynamic model of the student's knowledge and misconceptions. The model starts by scoring the predictions table, in which the student is asked to predict the qualitative changes caused by a perturbation [Rovick 1992]. The model is updated with every student input. **Version 2 uses a combination of overlay and bug library approaches, where common student misconceptions were found and catalogued by analyzing the tutoring tran-**

scripts. Version 3 will try to derive and incorporate more global information, such as overall student performance and signs of student strain.

Task 5: *Generation of Natural Language Text*. This is the focus of the research, the earlier tasks provide the necessary context. Generation of an intelligent tutorial dialogue requires the system to produce questions, acknowledgments, summaries, and hints as well as explanations, so it involves novel problems in language generation. The system must be able to respond to ill-formed input, which is full of typing errors, spelling problems, ellipses and fragments, and errors in grammar. It must also respond to all initiatives raised by the users. Both natural language generation and understanding are based on the developed LFG. The LFG has been found to be an excellent framework for understanding ill-formed input, and its incorporation of psycholinguistic insight has helped researchers build the grammar rules for questions, hints, acknowledgments, and other fragments that most grammatical theories do not support at all.

Task 6: *Hinting*. Hinting turns out to be both more important and more difficult than anticipated. Researchers have developed a theory that successfully determines when to hint, but are still working on methods for choosing the context and generating the language [Hume 1993]. Hints are structurally complex, and in many cases hinting language serves multiple functions, simultaneously indicating an error and a way to repair it. It is clear that hinting requires a much larger and richer knowledge base than is needed by more didactic tactics. Analysis of tutoring discourse shows that hints fall into two major categories: one in which the tutor conveys information to the student (ci-hint), and another in which the tutor points to information the student is thought to possess (pt-hint). In both cases, the tutor intends that the student use the (conveyed or pointed-to) information to solve the problem.

Task 7: *Domain Knowledge Base*. Domain knowledge is essential in planning the tutoring session, understanding student language, modeling the student, and generating responses. The researchers developed five different versions of the knowledge base before arriving at one that can support the desired tutoring. An effort to parse a whole chapter about cardiovascular physiology, using Sager's Linguistic String Parser with the goal of producing frames for the knowledge base, taught researchers much about the language of cardiovascular physiology, but the frame-based knowledge base it produced was still insufficient. The first component of Version 3 to be completed was a much larger, three-level, object-oriented knowledge base that expresses many complicated interactions between types of domain knowledge. The new instructional planner moves between simpler and more detailed levels in order to find the material for a hint or an explanation.

Task 8: *Hypermedia and Natural Language*. This task involves experimenting with ways to combine hypermedia and language in teaching causal reasoning. Researchers at Rush Medical College have built a hypermedia tutoring system called A-Base that teaches causal reasoning about Acid-Base reactions. This program uses only canned text but supports experiments with positive and negative acknowledgments of various flavors. A hypermedia program that allows the student to con-

struct a concept map on the screen has been built. It can also be used to display portions of the concept map at various points during the tutoring process.

Task 9: *Tutoring Systems.* **Rush medical students have used the fully functional Version 2 of the system as volunteers. Anecdotal reports suggest that this program is helpful as a learning resource, although the students pointed out several important limitations. These fall into three broad categories: (1) stereotyped text generation, (2) failure to summarize at appropriate points in the dialogue, and (3) failure to understand student input.**

Potential Follow-On Research Further work is envisioned to investigate four issues:

- Determining whether a learning environment built on top of the knowledge base works better than CircSim-Tutor or not.
- Investigation of alternative tutoring strategies.
- Study of hinting tactics.
- Investigation of the effect of display of the concept map on tutoring performance.

9.4.2 CircSim-Tutor

CircSim-Tutor is designed to help students learn to understand and reason about the negative feedback system that controls blood pressure in the human body. It requires them to make qualitative predictions about the effect of perturbations of that system on several important physiological parameters.

Tutoring begins with the students providing seven predictions about the response of the system to a disturbance. These inputs are rich in information from which the tutor can infer the presence of student misconceptions or missing concepts or facts. The tutor thus identifies a set of topics or lessons to be pursued at some point in the dialogue. This "lesson stack" represents the set of possible topics to be pursued. When a lesson is identified, the tutor usually generates a plan for how to talk about this topic. Many of these plans are generated on the fly; others, particularly plans for remediating misconceptions, are pre-stored by the expert tutor, based on experience with students solving problems in this domain. There are multi-part hints, multi-part explanations, and directed lines of reasoning, in which the tutor pushes the student through a whole sequence of steps in the reasoning process.

The system asks questions constantly, both to move the dialogue on and to discover student misconceptions. It formulates hints sometimes as questions and sometimes as imperatives. If the student cannot come up with the right answer after a couple of hints, the system gives a brief explanation instead. Much of the material in the knowledge base was

added to enable the system to give better hints and explanations of the underlying physical concepts and to allow it to discuss the problem-solving process with the user.

Tutoring in this environment is most frequently error-driven and thus aimed at remediation. But even in the absence of prediction errors, tutoring proceeds with the goal of confirming the tutor's model of the student's cognitive state that is continuously under refinement.

Development Status The prototype system was called the Kim Prototype. Since then, Circsim-Tutor has evolved through Versions 1 and 2. Version 3 is under development.

Architecture Figure 28 on page 123 shows the major system components. In Version 3 the components are implemented as packages in the Common Lisp Object System (CLOS) and communicate with each other by means of messages. There are a number of common data stores that can be accessed by all of these modules: the Knowledge Base, the Lexicon, the Discourse History, the Tutoring History, and the Journal File (which captures the details of every tutor-student interaction).

Instructional Planner The Instructional Planner determines the system agenda, and with the aid of five sets of rules that apply at five different levels of the tutoring process, determines what the system will do next at every point. It consults the Student Model to decide which rules should be carried out. It communicates instructions to the Text Generator as to whether it should deliver a positive or negative acknowledgment, a hint, a question, a definition, or an explanation. It instructs the Screen Manager when it is time to display procedure descriptions or operating instructions.

Screen Manager The Screen Manager paints the Prediction Table on the screen, produces instructions and help messages as they are needed, and provides windows for natural language input and output. It collects all keystrokes and returns them to the Planner.

Input Understander Natural language input is passed to the Input Understander, which carries out spelling correction and parsing, and then produces a logical form. The logical form is then sent to the Student Modeler, as are the entries from the Prediction Table.

Student Modeler The Student Modeler analyzes the input with the help of the Problem Solver, which calls the Knowledge Base to obtain domain-specific information.

Problem Solver Each of the (approximately 70) problems describes a perturbation of the cardiovascular system and asks the student to predict qualitative changes in seven major physiological variables. The Problem Solver uses the algorithms and the data in the knowledge base to solve the problems that the system presents to students and queries from the other components. It records a trace of its reasoning process for use in the generation of explanations and hints.

Text Generator The Text Generator generates questions, brief explanations, acknowledgments, hints, and definitions in response to requests from the Planner. It receives an indication of what kind of contribution is needed and a logical form from the Planner. The Text Generator uses the logical form to generate an LFG functional structure

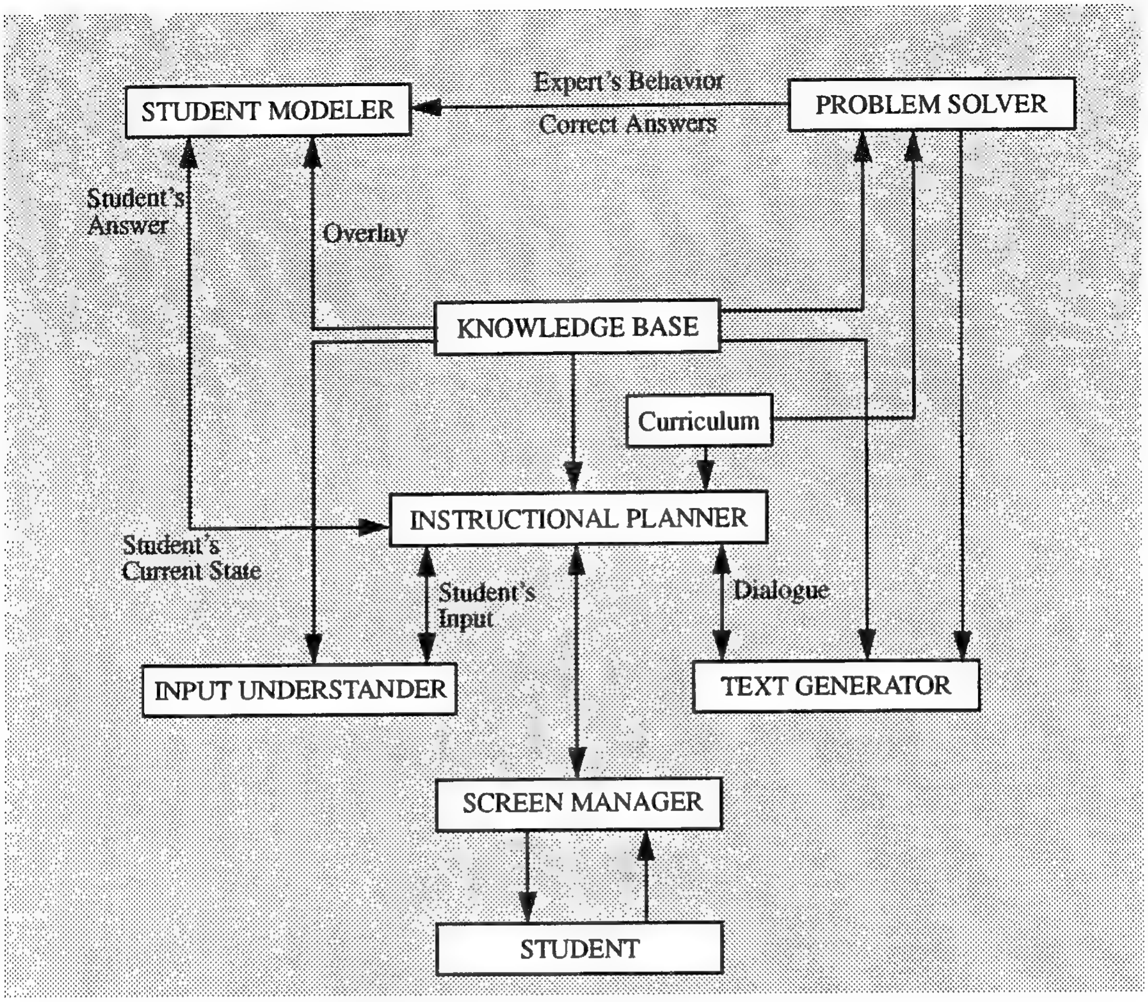


Figure 28. Organization of CircSim-Tutor

(f-structure), then translates that into a constituent structure (c-structure), and finally returns surface sentences to the Planner.

Evaluation Status A number of preliminary evaluations have been completed. The evaluation scenario consists of a pretest, an experience with the system, a posttest, and a questionnaire. (The pretest and posttest were developed to evaluate human tutoring sessions.) The focus of evaluation is on comparing computer tutoring with Circsim-Tutor and human tutoring, against tutoring via the computer-aided instruction program CIRCSIM (this is a session of the same length spent reading a textbook).

The most recent evaluation was conducted in Spring 1994, using the current stage of Version 2. Five medical students completed all four implemented problems in less than two hours and also a questionnaire. Researchers are still working on a formal analysis of the results. The students all said that they enjoyed using the program and felt they learned a great deal. They did not complain that the language generated was repetitive, as did students who used a previous version (the inclu-

sion of lexical choice in the surface generation appears to have resolved this problem). They did ask for longer explanations. They also complained that the system failed to understand their input and did not give good feedback about how to alter it. The questionnaire responses mention other problems with the interface—too many clicks are required and the window organization is awkward.

These student comments and the analysis of the student-computer dialogues have motivated current efforts to improve the language understanding and surface generation in Version 2. The design of the screen manager for Version 3 has already been changed in response to the student comments.

Operating Environment

Version 3 is being implemented in CLOS, the object-oriented version of Lisp. It runs on Macintosh computers SE-30 onwards. Development was largely carried out on a Macintosh IICi, but the system was ported to SE-30 and IIs machines for the Rush Medical College environment. It requires 5MB main memory. The system has also been demonstrated on Centris, Quadra, and portable Macintoshes. Development of a Windows (PC) version is on hold, pending resolution of serious bugs in the Franz version of the Procyon Lisp compiler.

Future System Development Plans

The researchers plan to build a more general tutor that can help students handle a whole range of negative feedback problems. They want to broaden the capacity of the natural language understanding and generation components. They also hope to improve the content and language of the tutoring hints and to increase the range of student initiatives.

9.4.3 Participating in Reflective Dialogues in a Complex, Real-World Domain Project

The original goal of this project was to provide a computational model that would allow an ITS to participate in reflective dialogues in complex domains. This focus reflected the growing interest in teaching complex problem-solving tasks, using computer-based intelligent apprenticeship environments that combine "learning-by-doing" with post-problem reflection sessions where students review their own actions and compare them to expert behavior. Experience with the Sherlock system (see Section 2.4.8), an intelligent apprenticeship environment that trains avionics technicians to troubleshoot complex electronic devices, has shown that building a system to participate in reflective dialogues in a complex domain poses a difficult challenge. Like most all fielded ITS, Sherlock generates responses to reflective query by filling in and printing templates that have been written *a priori*, and are selected on the basis of a set of features that describe a discourse situation. Due to the complexity of the domain, there is frequently a large amount of information that is potentially relevant to the student's question. Experience with the system has shown that when the situation is even moderately complex, explanations generated by concatenating all the relevant templates are unnatural, difficult to understand, and potentially misleading.

Clearly, if computers are to realize their potential as a powerful tool for facilitating learning through reflection, models for effective reflective interaction must be identified. This research is aimed at identifying the types of questions students wish to ask in reflective interactions, and identifying the strategies for choosing what information to include in responses to students' questions and for organizing and presenting that information in a manner that is intelligible to students.

Programmatic Background This work is being performed under contract N000149IJ1694, awarded to the University of Pittsburgh. The Principal Investigator is Dr. Johanna Moore. The project began in May 1991 and is due to complete in April 1997. It is a 6.1 effort with a total funding of approximately $370K.

Planned Products Prototype explanation generator for the Sherlock ITS. General purpose discourse planning formalism.

Approach Within the context of the Sherlock domain, data will be collected on the way that a human expert responds to trainee questions and provides explanations. AI techniques will be used to emulate this performance. Discourse goals will be identified and the system's information about the situation will be appropriately selected, abstracted, and summarized in order to generate a response that meets the standards of human behavior.

Task 1: *Corpus Gathering.* To gather a corpus of naturally occurring reflective interactions, researchers will perform a study where the Sherlock tutor is replaced with a human tutor. **This task has been completed.**

Task 2: *Corpus Analysis.* The corpus of naturally occurring reflective interactions will be analyzed to determine the important features that distinguish human-produced explanations from computer-generated utterances. This includes identifying the following:

- The types of questions students ask during reflective dialogues.
- The types of content that human tutors include in response to the various types of questions students ask.
- The rhetorical strategies for organizing this content into explanations that are intelligible to students.
- The linguistic cues that human tutors use to signal the structure of the domain information being conveyed, as well as the rhetorical organization of the text.
- Other strategies that human tutors use to manage the complexity of explanations.

A preliminary analysis of the corpus has been completed.

Task 3: *Identifying the Domain Knowledge and Explanation Knowledge Required by an Explanation System.* Implementing an explanation system also requires that the researchers identify the types of knowledge that the system must provide in its knowledge base to support the types of explanations given by human tutors. This knowledge falls into two categories:

- Domain Knowledge. Knowledge of how the domain works and strategies for solving problems in the domain.
- Explanation Knowledge. This includes discourse strategies for answering the range of questions students ask during reflective interactions in this domain, heuristics for selecting among multiple explanation strategies when they are available, general linguistic knowledge about how to organize explanations, how to refer to domain entities, etc.

Progress has been made in identifying knowledge requirements from both categories.

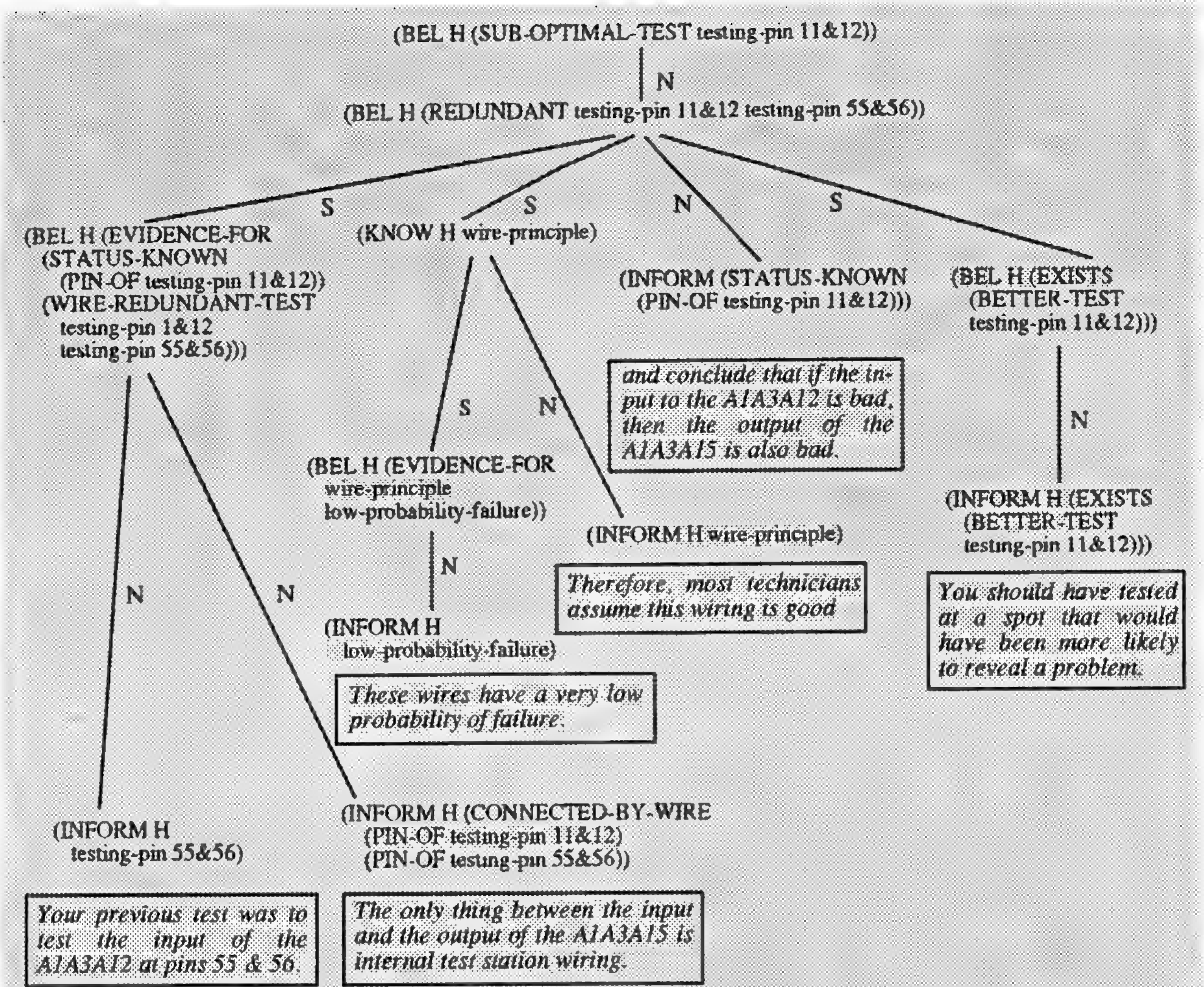
Task 4: *Implementing a Computational Explanation Generation System.* Using what is learned from the previous steps, researchers will implement a computational explanation generator that uses the identified strategies to provide explanations that have the properties observed in human tutor's explanations. **This work has begun with extending previous work on text planning. Figure 29 shows a example of a generated text plan, and Figure 30 on page 128 shows the system architecture for the explanation generation system.**

Task 5: *Evaluation of Computational System.* When the computational system is sufficiently robust, the effectiveness of generated explanations will be evaluated using real users on the job. The system will be programmed so that the each capability in the model can be enabled or disenabled. This will allow them to systematically evaluate each capability, and combination of capabilities, to determine which features of the explanation capability lead to the biggest improvements in the effectiveness of the training system.

Potential Follow-On

The researchers have proposed two major thrusts for extending this work. First, completed work has illustrated the need for more detailed linguistic analysis. Researchers will conduct three detailed linguistic studies of the corpus of reflective dialogues aimed at identifying the factors that influence the use of discourse markers, reference to prior explanations, and the generation of referring expressions. Each of these studies will yield rules that can be implemented and evaluated in the explanation generator for Sherlock.

Secondly, researchers have identified the need, in the longer term, for a new text planner that can support the features they have identified as important for a system that participates in natural reflective interactions. In particular, this text planner must be able to plan to achieve conjunctive goals, for example, "make the hearer believe that testing pin X is bad," and "make the hearer believe that testing pin X is similar to the previously explained test of pin Y." This requires moving from the current linear planner to a more sophisticated formalism which will allow the handling of goal interactions that arise when planning to achieve multiple goals. The researchers are in the process of extending a state-of-the-art AI planner for this purpose.




```
Operator 1:
EFFECT:         (BEL USER (SUBOPTIMAL-STEP ? current-meas))
CONSTRAINTS:(AND (MEAS-TEST ?current-meas)
                 ((OCCURS-BEFORE ?prev-meas ?current-meas))
                 (NEQUAL ?current-meas ?prev-meas)
                 (SATISFIES-GOAL ?current-meas ?goal)
                 (SATISFIES-GOAL ?prev-meas ?goal))
NUCLEUS:        (BEL USER (REDUNDANT ?current-meas ?prev-meas))

Operator 2:
EFFECT:         (BEL USER (REDUNDANT ?current=meas ?prev-meas))
CONSTRAINTS:(NAD (MEAS-TEST ?current-meas)
                 (MEAS-TEST ?prev-meas)
                 (CONNECTED-BY-WIRE (PIN-OF ? current-meas)
                                    PIN-OF ?prev-meas))
PRECONDITIONS:(KNOW USER wire-principle)
NUCLEUS:        (INFORM USER (STATUS-KNOWN (PIN-OF ?current-meas)))
SATELLITES:     (AND ((BEL USER (EVIDENCE-FOR
                            (STATUS-KNOWN (PIN-OF ?current-meas))
                            (WIRE-REDUNDANT-TEST ?current-meas ?prev-meas))))
                 ((BEL USER (EXISTS (BETTER-TEST ?current-meas))) *optional*))
```

Figure 29. Text Plan for Justifying the Tutor's Assessment of an Action

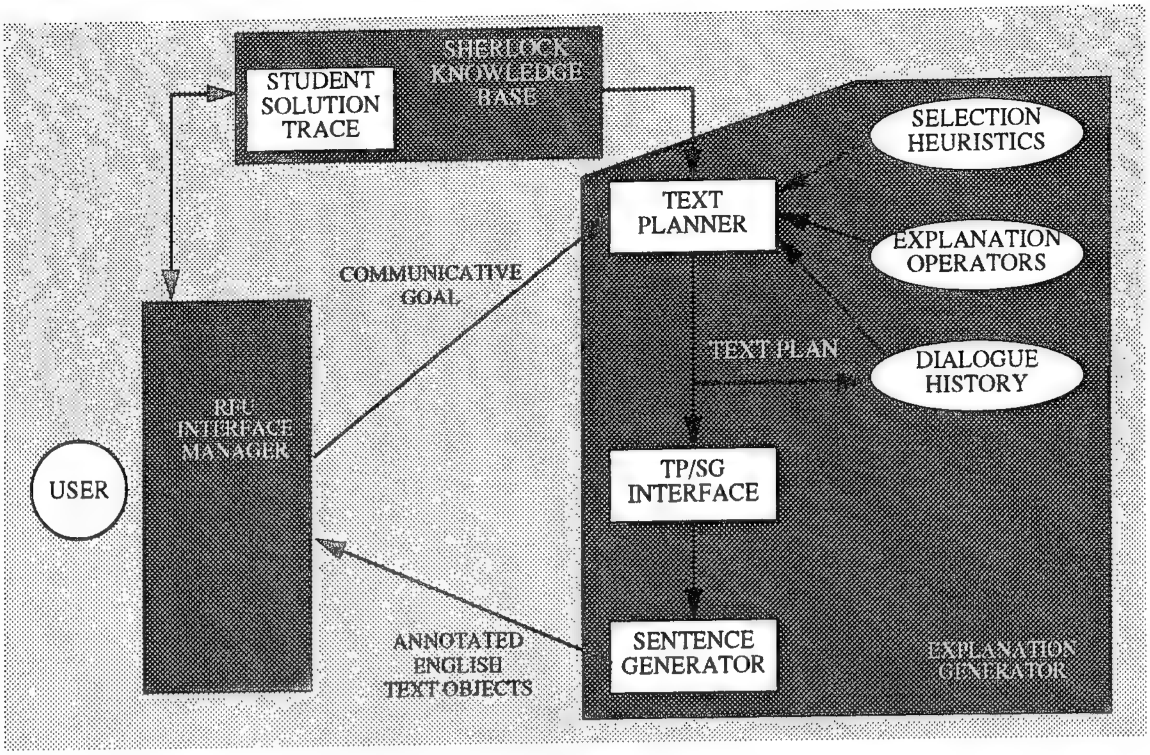


Figure 30. System Architecture of Intelligent Explanation Generator

9.4.4 Development and Assessment of Alternative Tutoring Strategies for an Intelligent Mathematics Tutor Project

The goal of this project is to develop a military version of an intelligent tutor for schema-based solution of algebra word problems and to experimentally determine which instructional strategies will maximize its effectiveness. This tutor, called the Tutoring in Problem Solving (TiPS) tutor, will be based on a previous system, called TAPS.

TiPS will be designed to support particular curriculum objectives and implemented within a training context based on cognitive apprenticeship. Although the TiPS architecture will be general enough to support problem solving training in different subject domains, the initial implementation will focus on complex arithmetic story problems as context for training relatively general problem solving skills. The first version of TiPS will support the following learning objectives:

- Understanding of arithmetic schemas: explicit representations of conceptual systems shown by cognitive research to underlie human understanding of basic mathematics.
- Strategic planning: the ability to select and organize schemas into solution steps that achieve an overall conceptualization and solution to a complex problem.
- Comprehension monitoring: the tendency and ability to be aware of one's own level of understanding and to recognize when needed information is missing.
- Systematic error-checking: the habit of checking one's problem-solving performance for avoidance of careless errors.
- Supporting beliefs: the maintenance and use of beliefs associated with good problem solving, such as the belief that problem solving requires persistence and that problem solving is supposed to be challenging (not easy).
- Selective encoding: the ability to identify important information and exclude extraneous information from the problem statement or situation.
- Strategic search: the ability to search for needed problem information within organized standard sources, such as dictionaries, reference materials, and manuals.
- Strategic integration: the ability to integrate and combine information from disparate sources to achieve a unified problem representation and solution.

- Cognitive negotiation skills: the ability to work with others to negotiate a problem solution based on mutual understanding of the problem.

An additional objective to be supported by later versions of TiPS is multiple representational levels, that is, memory management with multiple levels of problem representation.

The tutor is based on the schema theory of mathematics word problem solving that was developed in earlier ONR-sponsored research. In addition to this strong theory base, the tutor incorporates a novel fuzzy logic approach to modeling student knowledge, more effectively dealing with the inherent uncertainty of this process. Current tutoring systems mainly incorporate rather arbitrary decisions about the instructional strategies employed because a research base for those decisions is lacking. This project focuses on experimental studies of alternative strategies. In addition, the behavior of expert human tutors in this subject matter will be studied. The fact that a tutor has already been built provides a cost-effective situation for pursuing these issues of instructional strategy.

Programmatic Background

This work is being performed under contract N000149310310 awarded to the University of Wisconsin. The project began in February 1993 and is due to complete in January 1996. It is funded under 6.2 money, with a total funding amount of approximately $523K.

Researchers at the Department of Educational Psychology, University of Wisconsin-Madison, are supported by researchers from the Center for Mathematics and Science Education, San Diego State University, and the Department of Computer Science, Florida State University. The Principal Investigator is Dr. Sharon Derry.

Planned Products

TiPS tutor.

Prior Work

TAPS is a computer-based instructional tool designed to help students develop metacognitive skills and awareness. Examples of metacognitive processes targeted by TAPS include strategic planning and comprehension monitoring. It is implemented in adult education centers and continues to be used by researchers at the University of Wisconsin-Madison to support research into cognitive issues.

The new implementation incorporates a schema theory based on the work of Sandra Marshall, which was conducted under ONR auspices.

Approach

Human tutors will be studied to determine how they interact with and evaluate students engaged in problem-solving routines not previously examined in tutoring studies. Tutoring methods will be implemented in the already existing tutoring system using various AI techniques. Resulting system variations will be comparatively evaluated in instructional experiments.

Progress

TiPs version 1.0 was released in January 1994. Fifty multi-step work problems representing Naval on-the-job problem situations have been entered into the system and data from high-ability college students are being collected to provide normative solution paths for these problems. A parallel set of civilian everyday life problems is being developed. A dissertation experiment is underway comparing

two approaches to adult problem solving training that reflect two alternative hypotheses concerning major sources of difficulty. An agreement has been made with the Madison Area Technical College to use the TiPS system for remedial instruction, permitting large-scale data collection.

9.4.5 Tutoring in Problem Solving (TiPS) ITS

Students typically interact with TiPS by first selecting a story problem exercise from a problem bank representing a range of basic math concepts, real-world contexts, and levels of challenge, and then by analyzing and solving the problem using the TiPS graphics interface. Figure 31 on page 132 shows a sample TiPS screen.

TiPS students are expected to study worked examples and solve problems for the purpose of acquiring the generalized problem solving skills of the TiPS curriculum. The system developers envision that students' study and problem-solving activities on TiPS will be guided by a combination of system and human tutoring. For the current version of TiPS, the ideal arrangement is to implement TiPS in a learning center where there are knowledgeable, trained tutors who can work closely with beginning students until their problem-solving and system skills have reached a level where tutorial support can be phased out. For more advanced students, it is sufficient to have human tutors available on an "as-needed" basis to answer questions and provide other kinds of help. Centers organized in this manner employ a cognitive apprenticeship model, characterized by strongly scaffolded problem solving in early training, followed by gradually fading support in later training. As newer releases of TiPS are made available with more powerful mentoring capabilities, less time and attention will be required from human tutors, even in early stages of training, although some degree of human interactive support will always be desirable.

Regardless of whether tutorial scaffolding is provided by humans or by the system itself, the style of the tutorial support given can vary widely, ranging from the totally unobtrusive tutoring style where the tutor intervenes only at the student's request for help, to a highly controlling style, where the tutor intervenes with every error so that the student is forced to stay on an expert path. Here metaphors are used to help train TiPS trainees, asking them to examine, develop, and alter the metaphors they use for interacting and guiding students. The tutor takes a series of roles:

- Stage 1: Modeler, demonstrator, open expert system.
- Stage 2: Collaborator, Socratic teacher, demonstrator.
- Stage 3: Coach, cheerleader.

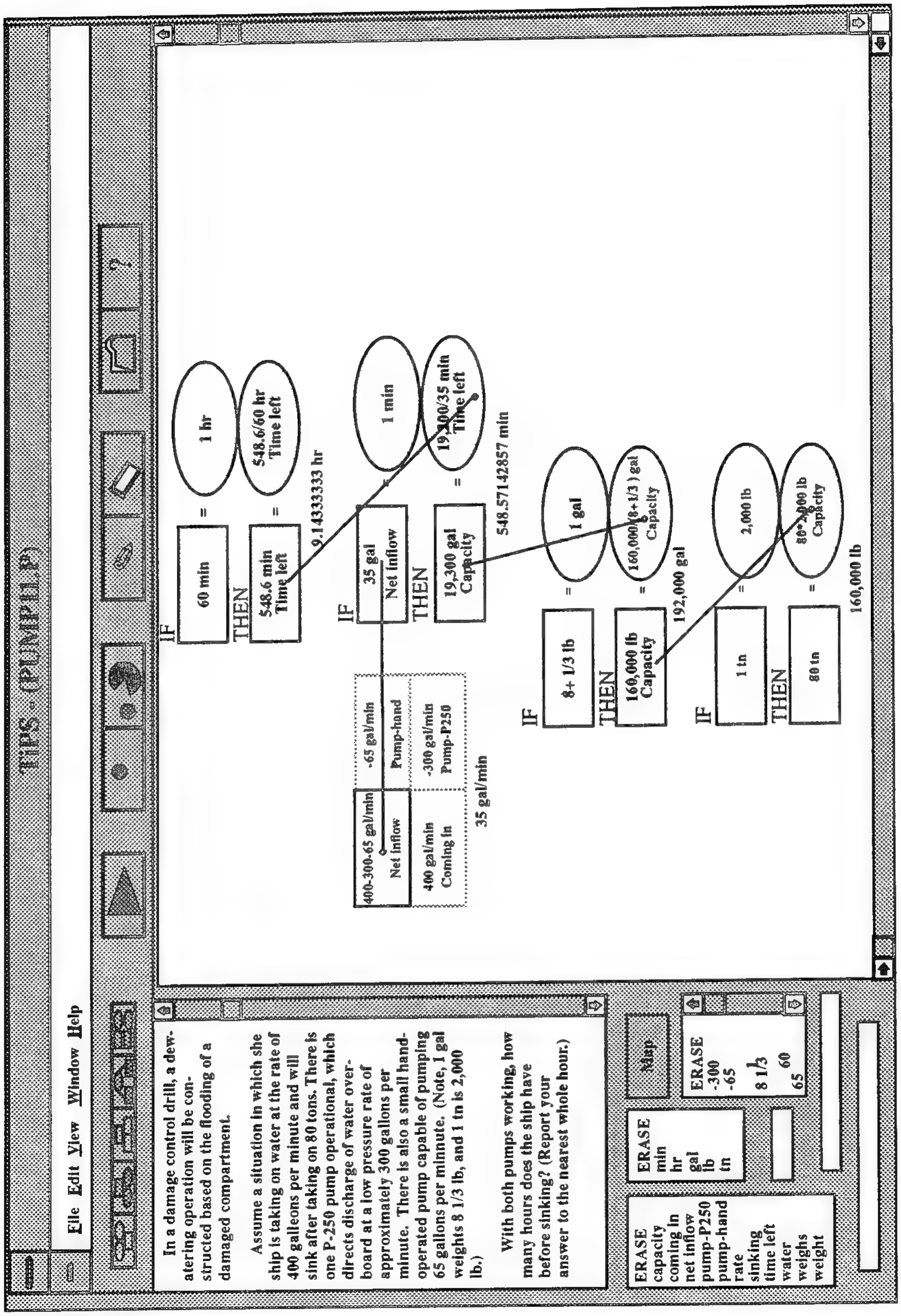


Figure 31. Sample TiPS Screen

A convenient and simple way for instructors to implement this three-stage cognitive apprenticeship model in TiPS is to tutor with think-aloud worked examples. Such an example has two phases: demonstration and problem solving. In the demonstration phase, the student observes while the tutor solves a problem, thinking aloud to make underlying cognitive processes explicit. Since the instructional session should have specific skill objectives, those particular skills are emphasized in the demonstration and think-aloud dialogue. In the problem-solving phase, the student or student group is given one or more problems to work, similar to the one demonstrated by the tutor. The students are asked to work their own example, trying to model the methods and behaviors demonstrated by the tutor. If the tutor wishes, students may be asked to think aloud about their own problem-solving processes, as this will help making their thinking processes public so they can be guided and corrected by the tutor. During this phase, the tutor provides scaffolding and fading as necessary, directing primary emphasis during the session on the target skills for that session.

While learning through problem solving, users should be encouraged to look back and reflect upon the worked example that has illustrated appropriate use of the target problem solving skills. This "looking back" facility is possible on TiPS through the multiple windowing and recording capabilities of the system. On TiPS, the demonstration phase of worked examples routines can be carried out by human tutors and captured by recordings that permit playback of both the voice and screen movements that the tutor demonstrated. Multiple windowing allows users to then solve their own problem in one window, while periodically referring back to an expert worked example being played and replayed in another window.

The TiPS recording facility also makes it possible for students to access and study think-aloud worked examples in the absence of a human tutor. Using the recording facility, a researcher or instructor can carefully design permanent dynamic worked examples and enter them into the system.

Development Status	TiPS is under continuous development as a number of upgrades are being planned and implemented. TiPS version 1.0 was released in January 1994.
Architecture	The TiPS architecture is shown in Figure 32 on page 134. The five major components are the student interface, problem bank, student record, student model, and a tutor. System development work to date has focused on the overall design, the student interface, and the problem bank.
Evaluation Status	TiPS is being used and evaluated by continuing education students enrolled at Madison Area Technical College, WI. Plans for implementing the system at a naval base are now being made. The site for this research will likely be Pensacola, FL.

Operating Environment

486 computers.

Future System Development Plans

Future versions of TiPS will be capable of tutoring students with less human intervention. System features currently being added to TiPS include the following:

- Local student modeling, allowing for context-sensitive performance feedback.
- Global student modeling permitting system-controlled, curriculum-level planning and summary evaluative feedback on progress toward objectives.
- System-controlled tutoring.
- Additional graphics interface tools prompting various levels of conceptual analysis.

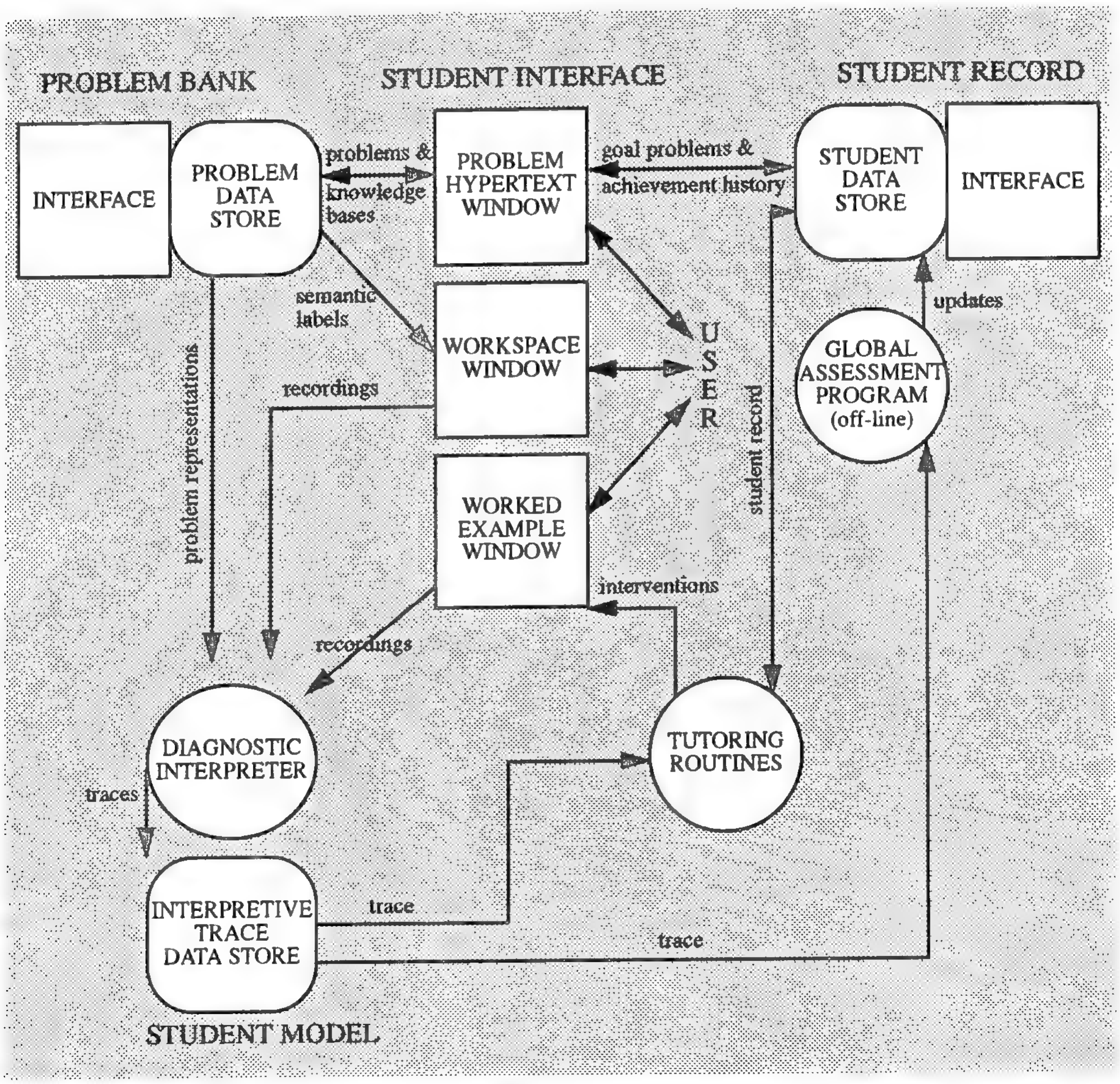


Figure 32. Planned TiPS System Architecture

9.4.6 Forecasting Training Efficiency and Transfer Effectiveness: An Automated Cognitive Approach Project

To date, cognitive models which result from a detailed cognitive analysis have been applied to the development of intelligent tutors, the prediction of human performance in a task, and the prediction of the time to learn and degree of training transfer. Given the breadth of application of cognitive task modeling techniques, it is quickly becoming a core technology for advanced systems development. However, the benefits of this technology can only become fully utilized when the cost to develop such analyses and their associated models has been reduced. The objective of the project is to extend the current version of a cognitive analysis tool (CAT) which elicits Goals, Operators, Methods and Selection Rules (GOMS)-like models of procedural knowledge relative to system operations [Card 1983]. The extensions proposed shall produce a capability that allows for the forecasting of the time to train an individual to expertly operate a system and the degree of training transfer that can be expected between two systems.

Previous research suggests that stereotypic-required cognitive processes (such as the need to attend to something or the need to make a simple decision) can often be inferred from the overt behavior that subject matter experts typically specify when attempting a task analysis. In this project, a variety of Navy console operation tasks will be analyzed by the investigator to identify such stereotypic patterns, and the results will be used to extend the CAT. In addition, the extended system will impose style rules that have been suggested for writing cognitive models of novice learners and for making accurate predictions of required training times.

Programmatic Background

This effort is being performed under contact N000-14-94-1-0433. The prime contractor is Virginia Polytechnical Institute and State University's Management Systems Laboratories. Researchers from the University of Michigan and Carnegie-Mellon University will also participate. The Principal Investigator is Dr. Kent Williams from the Virginia Polytechnical Institute. The contract started in March 1994 and will complete in April 1995. The total funding amount is $160K.

Prior Work

Under previous ONR support, researchers developed an automated aid to guide users through the conduct of a GOMS-like cognitive task analysis. The target application for this initial research was the structuring of instructional curriculum and the generation of ideal student models for ITS.

Evaluations of the initial software tool showed that primitive operators associated with mental activities were not typically described by subject matter experts. Only 28% of the primitive mental operators of a given method were accurately described in subject-generated models. On the other hand, 85% of the physical operators were accurately described. These findings limit the applicability of the aid, in its current version, to descriptions of the rather explicit operations involved

in skilled performance. (For purposes of developing and structuring curriculum for intelligent tutors, the models generated by the current version were sufficient since the presence of mental operator descriptions in the content of curriculum is relatively nonexistent. The forecasting extensions, will, however, support ITS use.)

Planned Products Extended CATS software tool.

Approach The major tasks to transition the basic research and extend the current version of CAT to incorporate a training time and degree of transfer prediction capability are outlined below.

Task 1: *Implement Style Rules.* Integrate within the current GOMS modeling process, as operationalized by CAT, the style rules for transferring GOMS models into novice-level NGOMSL statements as proposed by Kieras [Kieras 1990]. The NGOMSL notation is a high-level representation of the lower-level production rule notation guidelines proposed by Bovair, Kieras, and Polson [Bovair 1990]. The style rules refer to requirements for converting GOMS models into an NGOMSL-executable system and to requirements for structuring GOMS methods into novice-level NGOMSL statements.

Task 2: *Inferring Mental and Perceptual Operators.* Modifications to the current CAT are required to acquire specifications of the full set of mental and perceptual operators. With such modifications to acquire the primitive operators, the style rules will be applied to yield novice-level NGOMSL rules for prediction of time to learn and degree of training transfer.

Task 3: *Develop C++ Routines to Compute Similarity Between Novice-Level Rules.* Existing heuristics for computing similarity between production rules will be translated from Lisp to C++ routines that can be integrated into CAT. Implied in the development of routines to implement the heuristic is the development of a limited syntax which will impose some minor constraints upon the user when describing goals and subgoals. Likewise, processes will have to be developed to identify the NGOMSL step statements that are potentially similar but have been described employing different verbal characterizations. These synonyms fall out from the task descriptions obtained in Task 2 and will be augmented by reference to synonym dictionaries. The output of the matching routines will provide the count of the number of new NGOMSL statements representing the total number of new underlying production rules from which the predictions can be made. The equation used to predict the training time or transfer time is: (30-60) minutes + 30 seconds x Number of NGOMSL statements to be learned [Kieras 1990].

Task 4: *Analyst-Defined Operators.* Analyst-defined operators are those which represent complex processes which defy description at the level of the known primitive operators. The interface for identifying an analyst-defined operator will be develop, ensuring that any further decomposition of the operator would be unproductive. This capability will incorporate the queries developed in Task 2 to isolate the lowest-level subsystem operators. A heuristic will be developed such that if the conditions of the heuristic are satisfied through a system of queries to the user, then the operation will be identified as analyst defined and not require any further decomposition.

Task 5: *System Evaluation.* This task constitutes Phase II and is proposed as an option dependent on the successful analysis, design, and development in Phase I. The evaluation involves building NGOMSL statements from the extended version of CAT for a targeted military system. Training time predictions would be generated by the enhanced CAT system employing the previously stated equation. Test subjects would first study detailed descriptions about how to perform tasks using the target system. The subjects would then be required to practice the methods studied on the target system and would be provided with appropriate feedback when an error was made. Subjects would be required to repeat the practice on the studied tasks until one error-free repetition of all methods studied and practiced was achieved. Performance measures would include (1) the time spent studying the initial instructions for each method to be learned, (2) the time spent practicing until each subject reached error-free performance, and (3) the total training time. Predicted performance would be compared to observed performance for mean total training time as a function of the number of NGOMSL statements to be learned, as computed by the extended version of CAT.

9.5 Future ONR Efforts

There are two ONR-sponsored efforts that are expected to start early this summer, the How People Learn from a Tutor Project and the Utility of Coaching Models and Improved Measurement Systems in an ITS Project.

9.5.1 How People Learn from a Tutor Project

Individual human tutoring is known to be a highly effective method of instruction, yielding achievement levels about two standard deviations about typical classroom instruction. Little is understood about exactly what features of tutorial instruction are responsible for these gains. In this project, researchers will model the cognitive processes and consequent learning in response to detailed tutorial actions. Tutoring will be conducted over a computer link and think-aloud protocols will be collected from learners as a source of data. The research will build upon prior research in which researchers have successfully modeled learning from a physics textbook. In addition, the generality of the results will be explored by modeling learning in a tutorial situation developed and intensively studied by other ONR-sponsored research where the topic of instruction is cardiac physiology.

Specifically, the objective of this project is to develop an abstract model which relates tutorial actions to resultant cognitive processes and learning effects in the student. This will be a two-stage model of the form

tutorial action --> cognitive process --> learning

The research will delineate the kinds of cognitive processes that are possible and typical as responses to tutorial actions. This amounts to a cognitive task analysis of the meta-skill of learning from a tutor. Researchers will study two different task domains (physics and physiology) and two different types of tutoring (human experts and a model-tracing tutor) to achieve generality. Although this analysis may provide the framework for further research, they may discover unexpected responses to tutorial actions. They also expect to find that students have surprising ways of avoiding the learning they were intended to do.

Researchers will ascertain which cognitive processes are preferred by more learners. These correlations will help to suggest which cognitive processes cause more effective learning and why. Such results will be important for developing a prescriptive theory of tutoring that can be used by designers of ITS. A second extension would be to develop "meta" instruction, which teaches students the best way to take advantage of a tutor's advice. Similarly, the efficiency of a tutoring system could perhaps be dramatically increased by adding brief training on how to use a tutoring system effectively, as part of the introduction to the system.

A small part of this work will be the development of a protocol collection and analysis tool based on digitalized audio. This facility should be generally useful, as protocols are a widely used type of data that are notoriously expensive to analyze.

Last, this research is a demonstration of a new way to evaluate tutoring systems. Usually, ITS are evaluated with pretests and posttests, which is useful for summative evaluation. The protocol collecting, transcribing, and coding techniques developed here should provide the kind of fine-grained methodology needed to find out exactly what the tutoring system is doing right and wrong.

Programmatic Background The University of Pittsburgh, Learning Research and Development Center (LRDC), has been selected as the contractor for this work and the Principal Investigator is Prof. Kurt VanLehn. A contract for this work is expected to be awarded in the near future. The project is scheduled to begin in May 1994 and complete in April 1996. It is funded as a 6.1 effort, with a total funding amount of $290K.

Planned Products Reports detailing the developed model. The model itself will be available for Lucid Common Lisp environments.

Prior Work ONR supported development of a computational model of learning from non-interactional, text sources (e.g., examples) during the period January 1988 to September 1993. The current effort will extend the model to cover learning from interactional sources of instruction, tutors.

Approach Three studies will be conducted in which students will be tutored via a computer link while they provide think-aloud protocols. Taxonomies will be developed for

both tutorial actions (such as feedback, hints, and demonstration) and for student cognitive responses (such as in response to negative feedback, diagnosing the error, fixing it, generalizing from the experience). The three studies will involve learning physics from an expert human tutor, learning cardiac physiology from expert human tutors, and learning physics from a computerized tutor.

9.5.2 Utility of Coaching Models and Improved Measurement Systems in an ITS Project

Artificially intelligent tutoring or coaching has been shown to be a highly effective method of instruction, especially for problem-solving tasks like diagnostic situations. The present project will explore the use of novel information processing measures to enhance the student modeling aspect of the intelligent tutor. In addition, it will provide information on the added value of both these information processing measures and artificially intelligent tutoring, as compared to standard computer-based instruction. Such information is needed as we move closer to widespread application of tutoring technology. As a side-benefit, this research is being done in a context that assures that the best version of the instruction will go into immediate application as training for Armed Services Vocational Aptitude Battery (ASVAB) administrators.

This project aims to determine whether the use of information processing measures of what a trainee is choosing to look at can enhance the quality of student modeling in an ITS and to determine the value added by this approach and/or by intelligent tutoring itself, as compared to standard computer-based instruction.

Programmatic Background This project is funded with 6.2 money at an approximate amount of $298K. RGI Incorporated has been selected as the contractor for this work and a contract is expected to be awarded in the near future. The work is due to begin in June 1994 and end in June 1996.

Planned Products Research reports. Computer-based instruction/ITS lesson on diagnosis of system problems for test administrators.

Approach A computer-based lesson in troubleshooting, the Computer Adaptive Testing-ASVAB system will be built in several different forms: standard computer-based instruction, intelligent tutoring, standard computer-based instruction enhanced with the use of information processing measures, and intelligent tutoring enhanced with the use of information processing measures. The effectiveness of these several instructional approaches will be comparatively studied.

10. NAVY PERSONNEL RESEARCH AND DEVELOPMENT CENTER

10.1 Mission and Role of NPRDC

Defense S&T Reliance (see Section 2.1) identified the Air Force as the lead joint service for intelligent training systems research. Accordingly, NPRDC co-located an operating unit at Armstrong Laboratory, Brooks AFB, in May 1993. This operating unit is responsible for the following:

- Conducting R&D on advanced training technologies.
- Facilitating technology transfer.

This NPRDC operating unit is proceeding to identify, develop, and implement cost-effective training technologies in four phases:

- Analyze AL/ HR training technologies, such as RIDES.
- Match relevant AL/HR products to identified training needs.
- Adapt and implement AL/HR products for Navy needs where appropriate and cost effective.
- Conduct R&D related to advanced training technologies while maintaining a technology watch at Armstrong Laboratory.

10.2 Summary of Past ITS Work

NPRDC has been involved in ITS R&D since the earliest days. It participated in the early Tri-Service development of SOPHIE, an ITS for electronic maintenance that guided much later work. While perhaps not itself classifiable as an ITS, the work on STEAMER, in particular, had a significant influence on later ITS development. Developed in the early 1980s, STEAMER provided a graphically animated environment, with an advisor, for training navy engineers in the operation of steam propulsion systems. The innovative part of STEAMER was the graphical orientation of the knowledge base. STEAMER is now in operational use by the Chief of Naval Reserve Forces. Currently, researchers at NPRDC are

developing a new version of STEAMER. called STEAMER-GT, which will provide instruction in gas turbine propulsion. (Further information on this work was not available for inclusion in this report.)

Later ITS efforts have included the development of such systems as the Intelligent Maintenance Training System (for corrective maintenance of the SH-3 blade fold system) and Computer-Based Memorization System (for memorization of hierarchically organized concepts).

10.3 Overall ITS Program

As indicated above, the undertaking of a large-scale R&D program is not part of the operating unit role. However, some limited R&D efforts looking at specialized issues are being conducted. These are described in the following section.

10.4 On-Going ITS-Related Tasks

This section discusses two current efforts.

10.4.1 Complex Cognitive Skills Project

The analysis, measurement, and enhancement of cognitive skills in Navy technical ratings is a persistent problem in designing and evaluating training. To obtain more accurate and prescriptive information about training adequacy, tools are needed to analyze cognitive skills, to assess operator proficiency, and to diagnose deficiencies. Accordingly, the objectives of this work are as follows:

- Objective 1: Develop tools to identify and analyze cognitive skills and mental models, and
- Objective 2: Determine the feasibility of using ITS as a means of training appropriate mental models.

This research will provide Navy training designers, developers, managers, and instructors with technologies to identify and analyze cognitive skills and to measure the effectiveness of existing and new instruction for teaching complex cognitive tasks. Improved analysis will enhance learning, retention, and job performance of personnel coming to the fleet from school assignments. Improvements, in learning and retention will significantly reduce fleet training requirements and improve readiness. The demonstration of intelligent system applications will increase the capability for teaching conceptual models in technically complex training.

Programmatic Background

This effort is sponsored by the Chief of Naval Research (Code 34), Exploratory Program Element 0602233N, Work Unit RM33T23.08. The project as a whole began October 1991 and is due to complete September 1995, with an overall funding of approximately $1.5M. For Objective 2 of the project, the timeframe is October 1992 to September 1995, and funding is approximately $375K.

The remainder of this discussion focuses on that part of the project that relates to ITS, that is, Objective 2. The work is being conducted in-house by the NPRDC (Code 131) and the Principal Investigator is Dr. John Schuler.

Planned Products

Requirements and design specifications for electronic warfare (EW) cognitive skills training tutor.

Approach

Objective 2 will determine the feasibility of using tutoring systems for instruction of cognitive skills. It is divided into three tasks.

Task 1: *Determine Training Support Requirements.* Researchers will identify performance problem areas in EW performance on the AN/SLQ-32, document training requirements for these domains, and determine the most appropriate instructional strategies for training. **This task has been completed. EW performance problems and associated training requirements have been documented. Candidate domains from the problem areas have been selected for tutor application based on developmental, implementation, and tutoring system preferences and constraints. These domains include (1) manipulation of the AN/SLQ-32 software to increase probability of detecting selected emitters, and (2) information gathering and dissemination, and are combined under a terminal learning objective of "Respond to requests for ESM information."**

Task 2: *Develop Alternative Training System Profiles.* Tutoring systems hold promise for fostering the development of models in a manner which leads to improved skill acquisition, retention, and performance. Researchers will evaluate existing tutoring system architectures for presentation and response attributes and instructional components and strategies that best fit the training requirements in the domains selected for investigation. They will further decompose the training requirements into the cognitive, declarative, and procedural components required for mastery of the domain-specific skills and knowledge. This will lead to further definition of situations for model application as well as the context in which to assess system effectiveness. **Criteria for tutor selection have been developed and the extent to which requisite attributes can be accommodated has been rated.**

Task 3: *Develop Adaptive System Functional Specifications.* Software specifications for a tutoring system will be developed. **Definition of prototype simulation requirements and determination of performance measures has begun.**

Potential Follow-On

One product of Objective 2 will be software specifications for a tutor specific to EW cognitive skills training. It is envisioned that this will be a product transitionable to 6.3 development.

10.4.2 Linking Cognitive Styles to Instructional Strategies for ITS Project

Differences in the ways in which individuals process information can affect both learning and performance. The desirability of tailoring instruction to take individual differences into account has been known for some time, but to date it has not been practical or feasible to perform such tailoring in military training. With the advent of flexible, adaptive intelligent tutoring approaches, however, it is now possible to develop training systems that adapt to individual differences in the aptitudes and learning styles of operators of such complex military applications as EW systems.

This effort will develop (in Phase I) and demonstrate (in Phase II) strategies for instruction and information presentation that adapt to intrinsic differences in the way that individual learners acquire and use information. These "cognitive styles" are stable, pervasive differences among individuals, as assessed through a variety of reliable instruments developed in prior research. Instructional strategies and information-presentation methods may be implemented by means of an ITS that provides training based on an assessment of the learner's cognitive styles.

The major innovation associated with this project is the systematic development of conceptual links between an individual's cognitive styles and methods for organizing and presenting information during training for specific EW tasks performed in the Combat Information Centers (CICs) on Navy ships. These conceptual links will be based on an in-depth task analysis that identifies the cognitive skills required to perform each task and develops hypotheses about the relationships between these cognitive skills and an individual's cognitive styles. Using these conceptual links, researchers will design an adaptive, user-centered ITS to train operators on selected tasks based on an assessment of their cognitive styles. The efficiency and effectiveness of the ITS will be tested through experiments in Phase II, and the conceptual links and the ITS design will be refined based on the results. This iterative process, guided by a conceptual framework, will result in a set of empirically validated links between cognitive styles, instructional strategies, and information-presentation methods.

Programmatic Background NPRDC has sponsored and funded Contract No. N66001-94-C-7013T-000, awarded to ALPHATECH, Inc. The Principal Investigator is Dr. Jean MacMillan. This project began December 1993 and is due to finish June 1994. The contract award for Phase I is not-to-exceed $50K.

Planned Products Implementation and Test Plan for an ITS specific to EW cognitive skills training.

Approach Figure 33 on page 145 provides an overview of the components and associated tasks required to develop and evaluate an ITS that is adaptive to differences in the

cognitive styles of individuals. The overall effort (Phases I and II) has three major thrusts: concept development, software development, and evaluation-measure development. Three corresponding types of components are shown in Figure 33 on page 145: (1) conceptual components that constitute the model of the user, (2) software or system components that compose the ITS, and (3) evaluation components that provide the means of assessing the success of the ITS. The tasks for Phase I are described below.

Task 1: *Evaluate Existing Models and Theories*. Figure 33 on page 145 shows the conceptual links that are necessary in order to develop an ITS that tailors instructional strategy and information presentation to an individual's cognitive style. To the extent possible, researchers will draw on previous models, theories, and results that link individual differences in learning to cognitive styles. However, many of

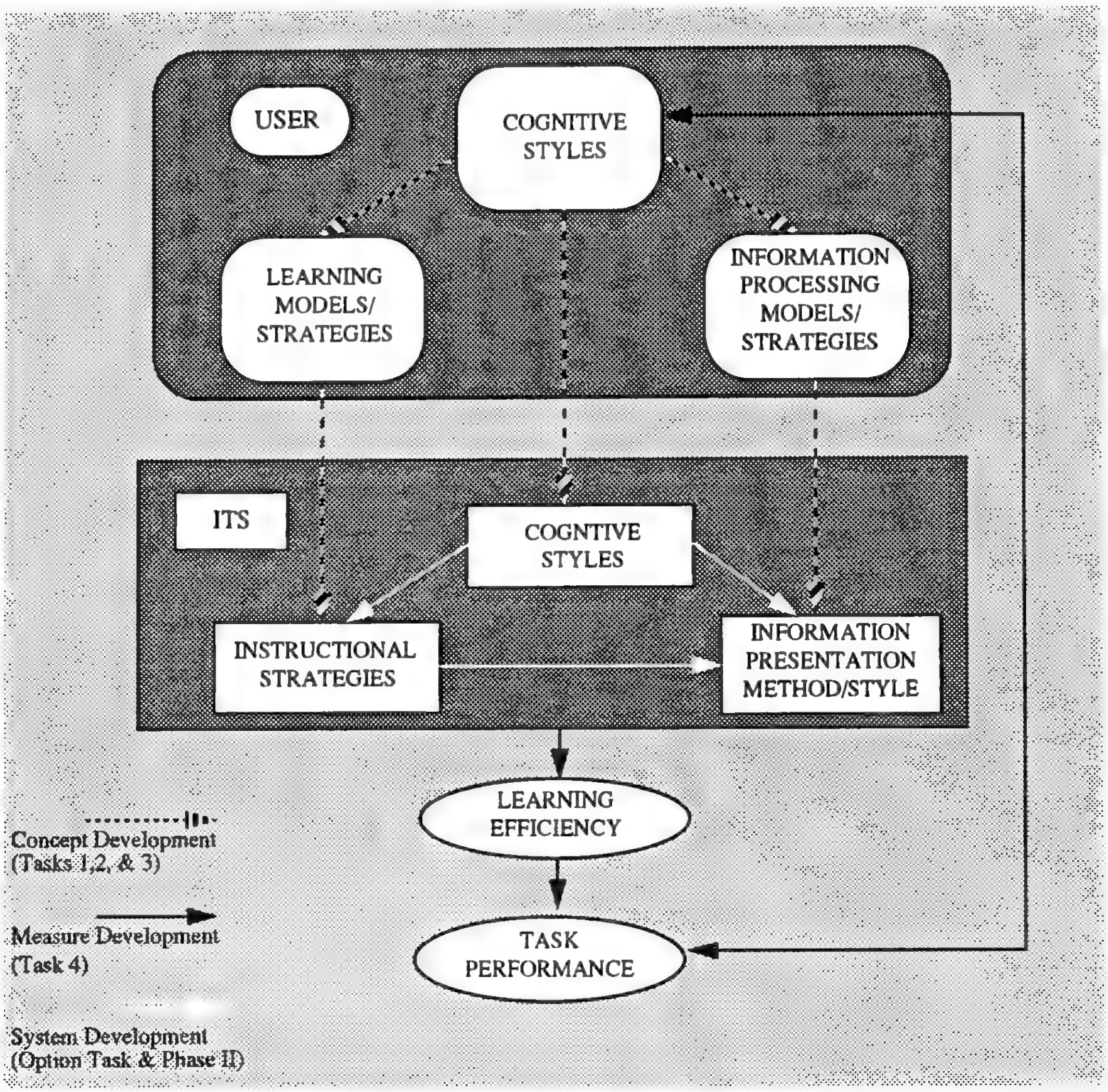


Figure 33. Overview of Technical Approach for Linking Cognitive Styles

the critical links in the figure are relatively unexplored in previous work. The first task in Phase I will be to expand this review, ensuring that all relevant prior work on the links between cognitive styles, learning models, information-presentation preferences, and instructional strategies are considered.

Task 2: *Develop and Refine Methodology.* Some of the conceptual links critical for the development of an ITS based on individual cognitive styles have not been well developed in previous research. In particular, the links between cognitive style dimensions, learning strategies, and information-processing strategies have not been well established. Researchers will adopt an iterative, common sense, trial-and-error technique to develop the conceptual links needed to adapt the instructional and information-presentation strategies of the ITS to the style of the individual.

Researchers plan to address the concept development task through a detailed analysis of the cognitive skills required for a specific set of tasks that are currently performed by Navy EW personnel. EW operators perform a number of monitoring, information fusion, and decision-making tasks that seem, based on previous research, likely to be affected by the cognitive styles of the individuals performing the task. Researchers will systematically analyze the functions and tasks of these operators and develop, for each task, hypotheses about the most likely relationship of cognitive styles to the task. Researchers will develop hypotheses in two areas: (1) the relationship between each cognitive-style dimension and the way in which the operator learns the task, and (2) the relationship between each cognitive-style dimension and the way that the operator processes information while learning the task. Both of the hypotheses will have implications for the design of the ITS.

The next step in the methodology will be to develop instructional strategies and information-presentation strategies for an ITS that trains operators to perform the task being analyzed, based on the hypotheses about the effects of cognitive style on learning the task. Researchers will eventually develop a set of strategies for the design of an ITS to teach a set of tasks (for example, all of the tasks composing one function of an EW operator) in a way that is adaptive to cognitive style. They will select the set of tasks that appears most promising for the design of a prototype ITS that demonstrates the concept of tailoring instruction to cognitive styles, and will develop measures for assessing the success of the concept.

Task 3: *Select and Analyze EW Tasks.* To demonstrate how an ITS can adapt to the cognitive styles of the learner, researchers will identify a particular set of EW tasks on which to focus, concentrating on tasks that call for the fusion of information from multiple sources in order to make decisions. They will perform an analysis of the selected tasks as described in Task 2. They will then evaluate the task requirements in conjunction with the cognitive-style dimensions described in Task 1 to identify the most relevant cognitive styles.

Task 4: *Develop Implementation and Test Plan.* The final task for Phase I will be the development of an Implementation and Test Plan that specifies how the concepts developed in Tasks 1, 2, and 3 can be implemented during Phase II to develop an ITS that trains EW operators to perform selected tasks. The Implementation

Plan will be in two parts—a plan for an initial prototype to demonstrate the concept and a plan for a more fully developed ITS. Associated with the Implementation Plan will be a Test Plan for assessing (in Phase II) the success of the strategies used in the ITS to adapt instruction and information presentation to the learner's cognitive styles. The Test Plan will specify the measures to be used and the hypotheses to be tested in the evaluation.

Potential Phase II Follow-On

The tasking of Phase II includes investigating the development of a dynamic learner model (within an ITS architecture) which incorporates assessed cognitive styles and from which instruction can be tailored. The incorporation of such information into a student model potentially could greatly increase the effectiveness of the tutor. Based on the results of the Phase II evaluation, the hypotheses about the links between cognitive styles and learning strategies for the tasks being studied will be refined.

10.5 Future NPRDC Efforts

This section provides currently available information on a project expected to start late 1994.

10.5.1 Damage Control Training in a Virtual Environment Project

NPRDC released a SBIR solicitation in the FY94 SBIR Program, Program Solicitation 94.1. The solicitation identifier is N94-123 and the category is Advanced Development; Training Devices.

> The objective of this project will be to demonstrate techniques for integrating damage control team training in a virtual environment. The resulting system is expected to include an ITS. The SBIR description is as follows:
>
> The skills for recovering from a shipboard casualty when under attack are critical to the survival of the ship. Training these skills is difficult since integrated team practice for Damage Control, Engineering, and Combat Systems personnel in a shipboard environment is not performed on a regular basis, and is not an experience that can be practiced to improve performance. Virtual environment technology may be suited for training teams in a simulated dangerous shipboard environment. The demonstration must support multiple participants experiencing a mass conflagration situation, real-time instructor intervention, and embedded instructional techniques.
>
> Phase I: Relate VE techniques to existing multimedia and instructional system development techniques to determine the methods of developing and evaluating team skills in a virtual environment. Identify the strengths and weaknesses of the VE approach, and estimate the options and relative difficulty of each aspect of the approach. Develop system specifications for a prototype VE capability for TSS team training.
>
> Phase II: Develop a prototype Virtual Environment Damage Control training scenario and evaluate its training effectiveness. The scenario must allow students to experience the consequences of their decisions, and should, with

practice, improve their responses to rapidly changing Damage Control, Engineering, and Combat Systems events.

Phase III: The prototype VE training system will be integrated into a Damage Control training course.

Commercial Potential: The technology has application in the private sector for police, fire, and emergency disaster training.

NPRDC expects to announce the contract award in early Summer 1994.

11. NAVAL AIR WARFARE CENTER TRAINING SYSTEMS DIVISION

11.1 Mission and Role of NAWCTSD

The mission of the Naval Air Warfare Center Training Systems Division (NAWCTSD), formerly the Naval Training Systems Center (NTSC), is to provide the following:

- The principal center for research, development, test, and evaluation (RDT&E), acquisition, and product support of training systems for Navy, US Marine Corps, Army, and Allies.
- Interservice coordination and limited training systems support for the Air Force, Coast Guard, Federal Aviation Administration, and non-DOD organizations.
- Technology and information transfer to other Government organizations and foreign governments in support of national goals.

There is no overall ITS R&D program.

11.2 Summary of Past ITS-Related Work

Previous work involved the continual monitoring of industry innovations in ITS and related technologies for incorporation into procured training systems.

11.3 On-Going ITS-Related Tasks

There are two tasks discussed here. Although the first task discussed is an SBIR Phase I effort that was completed in December 1993, it is included because NAWCTSD is currently considering whether to proceed with Phase II.

11.3.1 Low-Cost Knowledge-Based Tool for Rapid Prototyping and Development of ITS Project

The primary objective of this SBIR Phase I project was to demonstrate the feasibility of an inexpensive commercial tool for rapid development of education courseware based on the computer-based training and ITS technologies. The ultimate authoring shell

would allow a user, in a government training or education environment, to create his own ITS. The major tasks addressed during Phase I were the design and development of three knowledge-based components of an ITS: Teacher Module, Expert Module, and Student Profiler Module.

A mid-course between computer-based training and ITS was taken, making the best use of work in both, leading to Intelligent Computer-Based Training (ICBT). This approach provided two advantages:

- The strategy allows rapid generation of courseware, thereby reducing the time and labor involved in using a computer-based training system while taking advantage of the adaptable nature of ITS.
- By seeking the mid-course, researchers would get a leg up on commercialization and rapid deployment of the system. That is, the risks associated with research-flavored, purely ITS technology are mitigated by incorporating the commercially available computer-based training technology.

A proof-of-concept demonstration was developed that showed it is possible to marry the stable computer-based training technology with innovative and robust ITS methodologies to create an inexpensive ICBT environment on common platforms for rapid courseware development. It should be straight forward to incorporate multimedia capabilities into the Phase II prototype as researchers already have demonstrated the integration of text and imagery.

Programmatic Background This project was conducted as SBIR Phase I contract N61339-93-C-0070 awarded to LNK Corporation Inc. The Principal Investigator was Dr. Srini Raghavan. The project began in June 1993 and finished in December 1993. The total funding amount was approximately $50K.

Products Design of ITS development tool.

Approach During Phase I, three major tasks were performed.

Task 1: *Identification of General Functionality of the ICBT Development System.* **This task has been completed.**

Task 2: *Initial Design of ICBT Development Environment Architecture.* **This task has been completed.**

Task 3: *Development of Proof-of-Concept Demonstration.* **A proof-of-concept demonstration was developed in the context of night vision training. This demonstration did not include all the components of the ICBT shell designed in Phase I; the two most fundamental components that constitute the intelligent aspects of the ICBT architecture were selected for simulation. These components illustrate the advantages of using established ITS technologies to explicitly represent the knowledge about**

the domain content and the tutoring process. The selected components were the Topic Selection and Tutor Reaction to the Student Response components. The tools used to achieve the demonstration were COTS packages such as Toolbook and standard high-level languages.

Potential Phase II Follow-On

Based on their Phase I experience, researchers have concluded that they can develop a complete ICBT authoring environment for today's PC-compatible machines. The proposed Phase II activity will therefore focus on two tasks with the goal of ensuring successful commercialization for both Government and broader educational markets.

First, the researchers propose to develop the complete set of ICBT modules that run in the Windows environment. This constitutes a standard programming task as each of the prototype components was produced using COTS, as described above. These modules will then be integrated into a fully functional prototype ICBT development system with a projected price of less than $300.

Second, the researchers will use the functional Phase II prototype to create several complete courses, the content of which will be developed in collaboration with NAWCTSD training personnel. The development of these courses will serve two purposes. One of them is to provide NAWCTSD with a set of immediately usable courses. The second is to provide important input for the user issues that will be critical to the success and acceptance of ICBT technology, which is a prerequisite for commercialization. To meet naval training requirements, the ITS authoring system will be implemented on IBM PC or compatible machines.

11.3.2 Intelligent Tutor for Electronic Warfare Situation Assessment Project

Common applications of computer-based training have been in areas where the training objectives and domain are well defined, feedback and student progress can be easily presented and gauged, and the training can be conducted in a "classroom" situation. A logical progression for this technology is its application to on-the-job training where the training objectives, measures of training effectiveness, and knowledge domain are not well defined. Such systems are needed when (1) there is a lack of training opportunities, (2) there is a need for "realistic" training, (3) there are high consequences of mistakes due to poor training, (4) decisions must be made despite incomplete or deceptive information, and (5) sufficient procedures for job performance cannot be defined. Shipboard EW training represents one such domain.

One extension of computer-based training to make it useful in these more dynamic domains is to create a system that uses expert-defined, problem-solving skills, strategies or models, and compare them to those used by the student. Student models can be inferred on the basis of monitoring student behaviors and the use of probe techniques (such as verbal reports or questioning). Concurrence and divergence between the models, assessed as a

function of outcome (was the answer correct and was it gained using a process similar to that of the expert), serves as the basis for feedback and skill building. Such systems could be embedded within the operational context to meet "train like you fight, fight like you train" requirements. This new generation of training systems is referred to as Intelligent Embedded Trainers (IETs).

The goal of this ongoing program is to develop a standard, modular architecture for the development of IET for shipboard EW operators.

Programmatic Background This Phase I effort is being performed under a contract, N61339-93-C-0092, with JJM Systems, Inc., Intelligent Control Technologies Division. The effort began in July 1993 and is scheduled to complete in July 1994. The total funding amount for this work is approximately $272K. The Principal Investigator is Dr. Jonathan Gluckman.

Products Proof-of-concept system for an Electronic Warfare Intelligent Embedded Trainer.

Approach Initial program objectives were to develop a generic architecture for the development of IET and to demonstrate such a system for the SLQ-32 EW system. Original program goals included linking the IET to an actual SLQ-32 aboard ship and stimulating the embedded trainer using existing systems. After initial attempts to find sufficient data sources and ports to attach the IET to were unsuccessful, the program shifted focus onto the development of the trainer and demonstrating it as a stand-alone system. The goal to actually embed this system into the SLQ-32 has not been abandoned and more recent efforts have revealed additional information that suggest that this may still be possible. The demonstration of the stand-alone capability has the advantage of successfully demonstrating the concept of IET and the additional ability to develop and link with supplemental training devices that sufficiently emulate the actual equipment to provide intelligent training.

Potential Phase II Follow-On The Phase II effort will develop a prototype system. Phase II is expected to begin in July 1994 and continue till July 1995. The funding allocated for Phase II is approximately $228K.

11.3.3 EW Intelligent Embedded Trainer

Two operating modes for the trainer are provided to support acquisition, evaluation, and feedback on student activity, and a third specifically for training feedback. The first mode is a real-time, free-run mode. Using this mode, uninterrupted student performance can be monitored and later evaluated and fed back to the student for training using Mode 3. The real-time mode is provided for several reasons. First, it affords an opportunity for the trainer to be used unintrusively within the context of ongoing operations. This capability is particularly important for embedded systems in which there is little time for that system to be dedicated solely to training and where it is only a component of a larger group of

systems, as is the case with the EW operator who is only one of many persons who might be participating in a ship or battle group training exercise.

While the real-time mode is useful for gathering information about deductive activities, inductive processes are usually hidden from observable activity and can, at best, only be inferred from such activity. Thus, an interactive training mode in which several techniques to isolate student performance and gather information specific to inductive processes, in this case situation awareness, is provided. The interactive training mode uses probe techniques, or interruption analysis, to gather information related to inductive processes. Using this approach, the training scenario is halted and the student is asked to perform activities or answer questions relative to specific training opportunities. Another novel aspect of this trainer is the use of expert-defined rule bases as part of the Knowledgeable Observation Analysis Linked Advisory System (KOALAS) hypothetical model used to evaluate the scenario in real-time for training opportunities. The specific rule base used during any given training session is determined as a function of the student's choice of predefined training objectives at set-up time. This allows a very modular implementation that maximizes the ability to use different AI techniques to match training objectives and to conserve processing resources by only activating rules germane to the current training session. Moreover, this implementation allows training to be tailored to individual student needs and for all aspects of the training to be independent of the scenario used. From the stand point of the EW trainer, this provides the flexibility to use already established fleet training scenarios, and incorporate new scenarios as they become available, without changes in functionality or utility of the trainer, and to run in real-time exercises.

Once a rule set for detecting training opportunities is active, the scenario stops for training whenever appropriate conditions occur. At this point, the trainer monitors observable student activities (keyboards inputs, etc.) until such time as the student keys that he has completed the task. At that time, a situation awareness battery patterned from the Situation Awareness Global Assessment Technique [Endsley 1988] is implemented. The questions developed specifically focus on gaining information about the student's current cognitive model and his inductive processes. Cognitive feedback is then given which focuses on presenting information to the student on the correctness of his response, what the salient pieces of information were and how to access that information, and what actions should or could have been taken. After feedback is given, the student is returned to the training scenario and allowed to proceed until the next training opportunity is identified.

Upon completion of a training session, a student has access to a third system mode, the Training Debrief. This mode focuses on giving the student a composite "report card" of his performance during the training session, direction for remedial activity, and information about how his performance differed from that of the expert. After the composite feedback, more specific feedback related to each of the training opportunities is available for student perusal. To compensate for potentially substantial time lags between when a student might have completed the training session and when he views the debrief, facilities are provided to allow the student to view all critical displays and information as it appeared during the training session for each of the individual training opportunities.

Development Status

Proof-of-concept system under development.

Architecture

Critical aspects of the architecture include the use of a proven process model of human decision-making and flexible knowledge engineering and AI techniques in combination with structured training objectives, cognitive feedback techniques, and performance assessment and tracking methods.

The model of human decision-making is presented in Figure 34 on page 155. This model is embedded in the KOALAS process architecture shown in Figure 35 on page 155 [Barrett 1990]. The KOALAS approach, like the human decision-making model, distinguishes between inductive and deductive components of decision-making, and allocates deductive processes to the computer and induction to the human, thus capitalizing on the strengths of both the system and the human. While current generation computer-based training approaches can be successfully applied to problems that have little to no inductive components, complex training problems, consistent with the many military jobs, have many inductive components and, therefore, an architecture capable of capturing this quality of the training problem was required.

The KOALAS structure was embedded in the training system presented in Figure 36 on page 156. The system functions such that prior to a training session, the system loads from the database into the hypothetical model expert models comprising both deductive actions, in this case expert search patterns through different information sources, and inductive components, in this case the general view of the tactical situation (situation awareness and assessment). This model, once loaded, runs in parallel to the training scenario being played so that it remains current; the expected deductive activities and situation awareness are consistent with the current state of the scenario, which represents ground truth. The system then monitors the activities of the student, recording activities that are both consistent with and divergent from those expected based on the expert model.

Operating Environment

The proof of concept system runs in a DOS environment, coded in the C programming language. Graphics were produced using digital maps provided by the Defense Mapping Agency and Halo graphics. The rule bases are implemented using CLIPS, a Government developed and owned AI tool.

Future System Development Plans

Future work on the system will focus on increasing the rule bases, converting to a windows-based environment, and adding the capability to stimulate the system from scenarios being run on network training systems such as the Battle Force Tactical Trainer (BFTT)

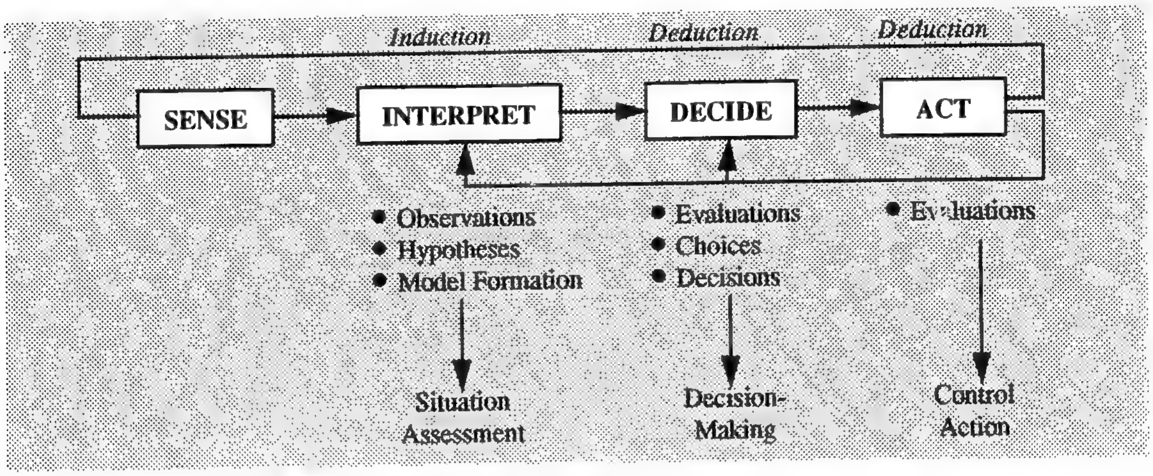


Figure 34. Human Operator Decision Process Model

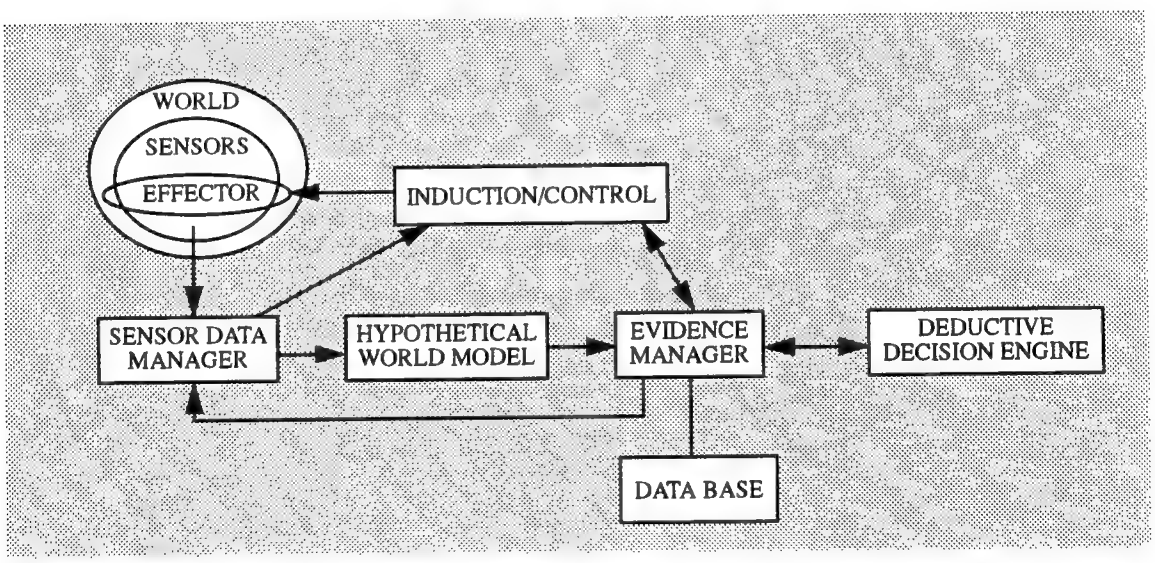


Figure 35. KOALAS Process Architecture for Intelligent Control Systems

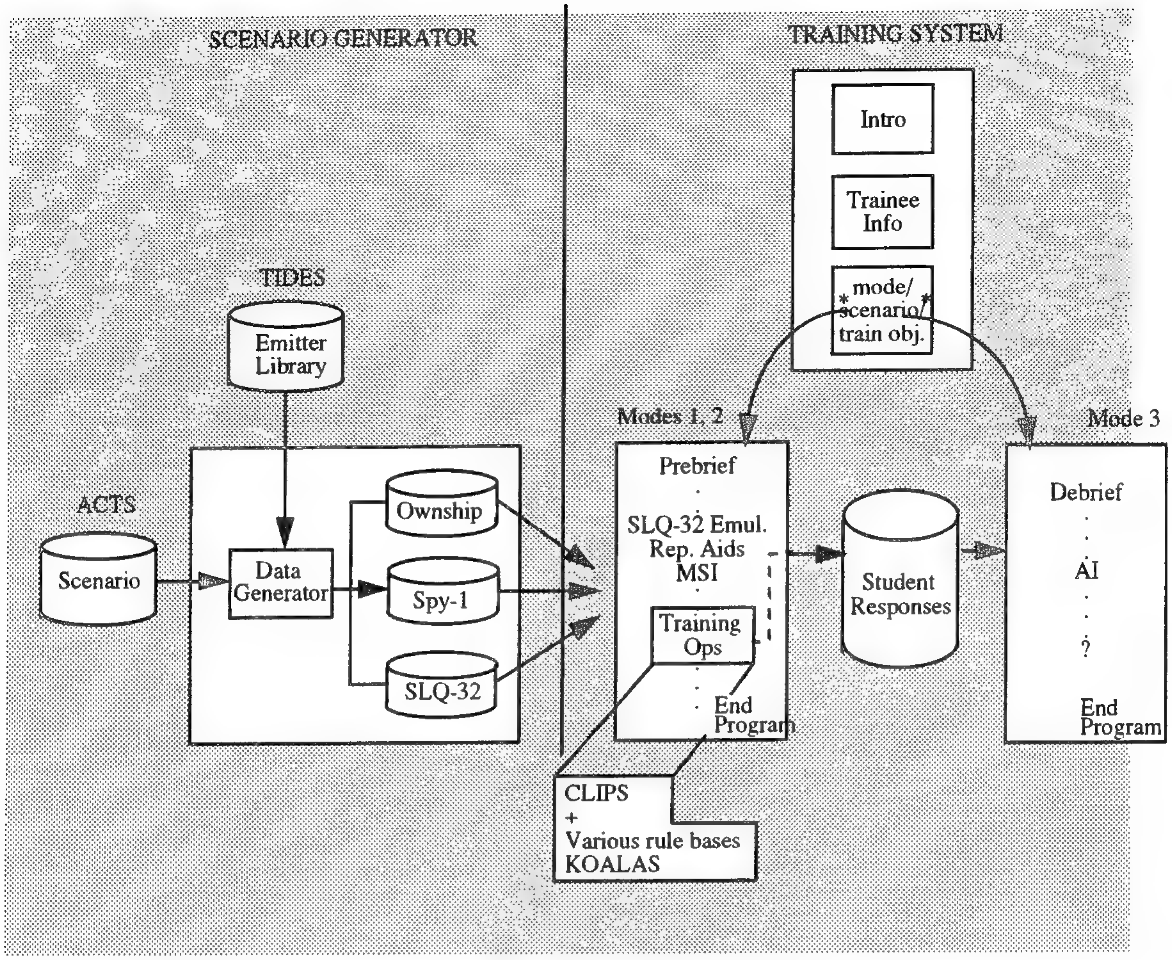


Figure 36. Overview of EW IET Architecture

12. SPACE AND NAVAL WARFARE SYSTEMS COMMAND

12.1 Work of Undersea Surveillance Division

Space and Naval Warfare Systems Command (SPAWAR), Undersea Surveillance Division, works in the area of active and passive anti-submarine warfare surveillance.

The SBIR effort discussed below is the first ITS R&D effort undertaken. Currently, there is no formal R&D plan and the possibility of future work in the area depends on the outcome of the current effort. SPAWAR expects to reach a decision whether to proceed with the SBIR Phase III in late Summer or early Fall 1994.

12.2 On-Going ITS-Related Tasks

12.2.1 Integrated Training Architecture in Support of Active Sonar Systems Project

The need for innovative Surveillance Towed Array Sensor System (SURTASS) training has increased significantly in recent years, in response to both major restructuring within the ocean surveillance community and changed mission requirements. Training requirements have become more specialized, the trainee population more diverse (in terms of both experience and geographic location) and the skills developed more perishable as opportunities to apply these skills in real-time operations become more sporadic. The SBIR work described here is dedicated to identifying, developing, and applying instructional technologies in a practical and cost effective way to meet these challenges. Phase I researched developments in instructional technology to select the most appropriate strategy. Phase II is in the process of building a prototype system that develops and applies that strategy to SURTASS Low Frequency Active (LFA) bistatic sonar analysis. Phase II work also includes the conceptual design of a comprehensive system to apply the technology throughout the ocean surveillance community. A Phase III follow-on effort is anticipated to develop the full-scale capability and meet the needs of the surveillance community.

Programmatic Background

This work is supported under SBIR Topic N91-052. The prime contractor is Digital Systems Resources, Inc. (DSR). Researchers from Virginia Polytechnical Institute (VPI) have also participated. The contract period for Phase I was August

1991 to February 1992. The Phase II contract period is from November 1992 through September 1994. The total funding amount is $555K.

Prior Phase I Work

The Phase I effort first assessed training system design and development methodology and then reviewed and analyzed both the advantages and disadvantages of shore-based versus on-board embedded training. Approximately 100 research reports were reviewed, organized, and analyzed to demonstrate how recent advances in cognition and instruction could support typical schoolhouse instructional activities in an on-board environment employing intelligent tutoring technology. The Phase I report reviewed two large bodies of research relative to curriculum design and strategies of instruction that can be implemented in embedded and on-board training environments. The final report also analyzed the advantages of shore-based training in light of newly developed technologies that can ensure those advantages in an on-board training environment, thus reducing the needs and cost of shore-based instruction.

Phase II work builds on Phase I by developing prototype demonstrations of both a qualitative modeling strategy for acquiring background knowledge of acoustic theory and a cognitive structuring and adaptive sequencing of curriculum for acquiring the procedures for device operation in an embedded environment.

Planned Products

The Phase II product will be the Multistatic Acoustic Receiver System (MARS) Trainer Demonstration System.

Approach

The approach taken seeks to maximize the value of each training hour by adapting the training sequence to the individual trainee. The selected training strategy alternates between diagnosis and training, bypassing areas of demonstrated trainee proficiency in favor of new or unfamiliar material. This results in efficient training for both the novice and expert. Additional emphasis is placed on creating a sufficiently structured and modular design to allow for local tailoring and revision by the user. The user, in this case, includes those responsible for developing, revising, coordinating, and administering training, as well as the trainee. The following tasks summarize the effects required to accomplish the objectives of the Phase II effort.

Task 1: *Bistatic Active Sonar Workstation Task Analysis.* A cognitive task analysis is needed to identify each functional step in the analysis process, associated console operations, and requisite knowledge elements. Based on this analysis, a set of device operation skill and knowledge goals can be selected as the scope of the prototype training. **This task has been completed.**

Task 2: *Define the Content Structure.* This task requires organizing the training material into a branching network of related concepts, skills, and knowledge elements. It will develop and apply a set of content development guidelines, to be applicable to any subject area, for implementation with the control strategy. **This task has been completed.**

Task 3: *Develop Content Material To Support Training Goals.* This task involves designing and creating the instructional frames for the prototype training system. Presentations and questions include text, graphics, and actual acoustic data displays, all

generated and displayed within the actual sonar display system. **This task has been completed.**

Task 4: *Design Instructional Control Strategy.* This task is concerned with defining the actual processes applied to the content structure to diagnose the trainee, adapt the training sequence, apply instructional strategies, provide feedback, handle trainee impasses, and provide review. **This task has been completed.**

Task 5: *Design Software to Implement the Instructional Control Strategy.* **This task is ongoing.**

Task 6: *Integrate and Test Software.* **This task is ongoing.**

Potential Phase III Follow-On The significance of the Phase II effort is in designing and demonstrating a practical application of intelligent tutoring technology that can be directly applied to any area relating to SURTASS sonar operator proficiency. Full-scale development of a comprehensive training system in Phase III would involve expanding and tailoring the courseware and improving the "user friendliness" of the training development and analysis functions, while reusing the content structure, control strategy, and control software previously developed.

12.2.2 DSR Training System

The tutoring processes within the Digital Systems Resources (DSR) Training System apply a control strategy to a defined content structure. The control strategy can be applied to any subject material that is appropriately modeled.

The training material must be organized as a network of related goals, concepts, facts, procedures, etc. The training system organizes this information as a tree structure consisting of individual elements referred to as nodes. The top-level node in the tree is the ultimate training objective. It will be broad and encompassing. Branching down from each node are component elements that need to be mastered in order to accomplish the node above. Nodes on the bottom of the tree (not broken down further) are called primitive nodes. Primitive nodes are relatively simple—such as a fact, definition, or specific action. Trees are developed according to guidelines that ensure consistency among nodal relationships.

Each node in the tree represents a complete lesson in a specific format. Each node consists of several elements, instructional components, and related parameters. The training author will be required to define the name, objective, presentation, problems, correct answers, and diagnostics for each node. Practice sets and diagnostic order are optional. Each node also includes parameters reflecting the node's position in the tree, the node's probability of success, trainee-generated comments related to the node, and indications of

the node's status with respect to the current trainee. These are defined and stored by the system.

The control strategy teaches the material by alternating between diagnostic and training processes, adapting the training sequence based on trainee responses and the tree structure. A depth of detail is built into the tree, yet only as much information as each trainee needs is actually presented. The system adjusts the instructional strategy applied when a trainee has difficulty with a specific problem, and a variety of processes provide review to confirm understanding and strengthen retention. Trainee performance data is recorded and saved such that patterns of skill development can be identified across large numbers of trainees. This information can be used to further refine the content material and structure.

In addition to the training technology employed, the DSR Training System will include features that enhance its versatility and utility for all users (training developers, and administrators, as well as trainees).

Development Status

The MARS Trainer Demonstration System is a prototype system under development. It demonstrates the training strategies and control processes of the DSR Training System, as applied to bistatic active sonar analysis.

Architecture

Major elements of the system are illustrated in Figure 37 on page 161. The MARS Trainer Demonstration System control logic interfaces with the MARS system display processor to generate training presentations that include actual acoustic data and system displays. A controlled level of MARS functionality is made available to the trainee, consistent with individual training objectives. The MARS Trainer Demonstration System operator interface is designed to be consistent with the "look and feel" of the host system operator interface.

Tutor Subsystem

The heart of the DSR Training System is the Tutor Subsystem. The Tutor Subsystem encompasses the instructional technologies employed and includes the selected content structure and all instructional strategy and control processes.

Developer

The Developer allows the user to create and modify the content structure, and includes an authoring tool for designing individual presentations, problems, and practice sets. The Developer will store training elements as nodes, incorporating linkages to other nodes to form the content structure. As the material is developed, the Developer will provide error checking to ensure that the rules of content development are not violated, and will ensure that nodes are properly specified.

Analyzer

The Analyzer allows the training coordinator and/or administrator to monitor the training performance of either an individual, a group, or the training course itself. It provides access to trainee-generated comments, and content-dependent and performance-based parameters. The Analyzer generates a variety of standard reports, and provides responses to specific data requests. Analyzer output reflects strengths and weaknesses in the content material, allowing the system to coach its own improvement.

Application Interface A clearly defined Application Interface allows the DSR Training System to be used effectively with a variety of to-be-learned systems, with minimum effect on the design and code of each one.

Evaluation Status The MARS Trainer Demonstration System will be used and evaluated in conjunction with traditional training methods to prepare operators for upcoming at-sea testing.

Operating Environment COTS equipment.

Future System Development Plans Future development plans include the specification of a system interface that will allow this training technology to be used with multiple systems, with minimal effect on the application code of each one, and the enhancement of training development and analysis functions.

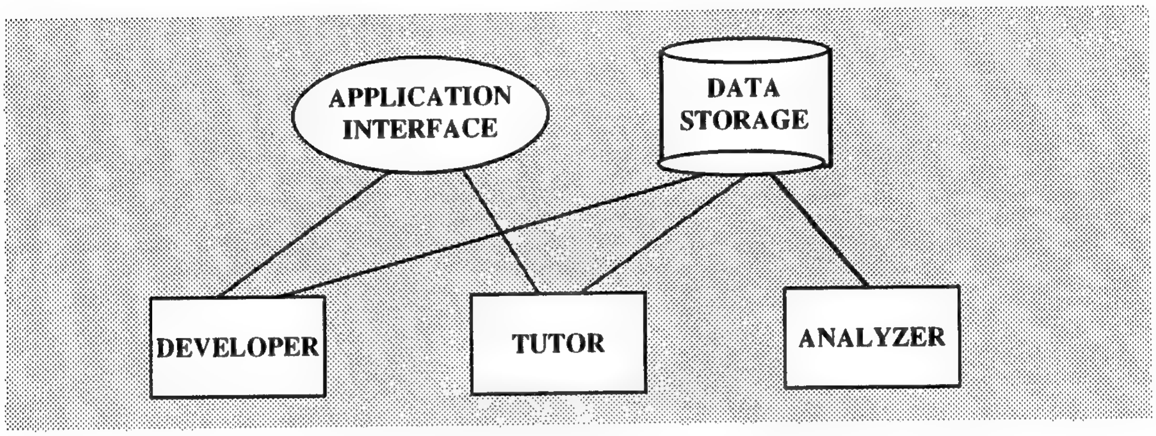


Figure 37. Major Elements of the DSR Training System

13. DEPARTMENT OF EDUCATION

ITS-related work is being sponsored by the Fund for Improvement of Postsecondary Education (FIPSE), part of the Department of Education (DOE) Office of Postsecondary Education.

13.1 Background to the FIPSE

Title X of the Higher Education Act, as amended in 1992, authorizes the Secretary of Health, Education, and Welfare to improve post-secondary educational opportunities by providing assistance to educational institutions and agencies for a broad range of reforms and innovations. To implement this authority, the Secretary of Health, Education, and Welfare established FIPSE, a separate organizational unit now within the Office of Postsecondary Education.

In the authorizing legislation, Congress identified eight broad purposes for which grants and contracts may be awarded:

- Encouraging the reform, innovation, and improvement of postsecondary education and providing equal educational opportunity for all.
- The creation of institutions and programs involving new paths to career and professional training and new combinations of academic and experiential learning.
- The establishment of institutions and programs based on the technology of communications.
- The carrying out in postsecondary educational institutions of changes in internal structure and operations designed to clarify institutional priorities and purposes.
- The design and introduction of cost-effective methods of instruction and operation.
- The introduction of institutional reforms designed to expand individual opportunities for entering and re-entering institutions and pursuing programs of study tailored to individual needs.

- The introduction of reforms in graduate education, in the structure of academic professions, and in the recruitment and retention of faculties.
- The creation of new institutions and programs for examining and awarding credentials to individuals, and the introduction of reforms in current institutional practices related thereto.

FIPSE administers grant programs as its principal means to achieve these purposes, and occasional targeted programs are sponsored which focus on particular priorities or concerns. Projects may received support for one, two, or three-year periods. Most funded projects are local improvements that continue beyond the period of Federal support; however, successful projects are generally designed to influence practice elsewhere.

13.2 FIPSE Program for FY94

The main activity of FIPSE is its Comprehensive Program, an annual grants competition designed to be comprehensive, action-oriented, risk-taking (in fostering new ways of achieving goals in secondary education), and responsive to FIPSE-indicated problems identified in the annual Agenda for Improvement.

Topics identified in the FY94 Agenda for Improvement are summarized in Figure 38 on page 165. It should be noted that each annual agenda provides guidelines intended to stimulate, not restrict, the thinking of potential applications. As can be seen in the figure, there is no specific R&D program for ITS technology in isolation.

Access, Retention, & Completion	FIPSE invites proposals to ensure that initial access to higher education is made more meaningful by improving retention and graduation rates without compromising academic standards, including retention of members of underrepresented minority groups in the academic disciplines beyond bachelor level.
Colleges and Schools	FIPSE solicits proposals that seek to improve the quality, accessibility, and retention rates of colleges and universities by helping them cooperate with elementary and secondary schools in the following areas: (i) the pre-service and in-service education of school teachers and administrators, (ii) articulation of curricula between schools and colleges, (iii) the strengthening of incentives for schools to offer, and for their students to excel in, sound academic programs.
Education & the Workforce	FIPSE seeks to encourage institutions of higher education to join with employers and secondary schools in the development of new models for integrating work and learning.
Curriculum Reform	FIPSE wants to support curriculum reforms at the undergraduate, graduate, and professional levels that will help student combine professional and technical expertise with an understanding of human diversity.

Learning-Friendly Campus Ethos	FIPSE welcomes proposals to make campus culture more conducive to academic progress by all postsecondary students.
Faculty Development	FIPSE solicits proposals to develop faculty as professionals by (i) improving the preparation for teaching of Ph.D. candidates planning careers in teaching at the postsecondary level; (ii) recognizing and rewarding effective teaching through appointment, promotion, and compensation policies; (iii) developing more effective methods of postsecondary instruction; and (iv) providing new opportunities for faculty to stay current with content developments in the broad range of areas they typically teach.
Financing & Educational Reform	Given the unusual financial pressures now facing many public and private colleges and universities, FIPSE welcomes proposals to experiment with new ways to maintain the quality and accessibility of education despite shrinking resources.
Dissemination of Successful Innovation	Recognizing that many national problems in higher education have been constructively addressed on some campus already, FIPSE invites proposals to disseminate properly researched and documented solutions from their original sites to other institutions.

Figure 38. Topics Included in FIPSE FY94 Agenda for Improvement

13.3 Summary of Past ITS-Related Work

Information on past efforts is not available.

13.4 On-Going ITS-Related Tasks

FIPSE is currently supporting two ITS-related tasks, the Computer Lessons for Written Harmony Project and the Computerized Sight-Singing Lessons with Intelligent Feedback Project.

13.4.1 Computer Lessons for Written Harmony Project

The study of four-part choral style forms the skill basis for the study of more advanced musical concepts such as composing, arranging, and analysis. In many ways, it is like a board game in which pieces (musical notes) can be moved about the board (musical staff) within the constraints of an historically evolved rule set. The implications of a decision made early in a progression (sequence of chords) may not become obvious until many chords later when an impossible situation is reached. The rule base is large and varied, with rules governing such facets as chord spelling, chord doubling, chord spacing, voice range, and connection procedures. Learning to write in this idiom takes extensive practice and careful, on-the-spot, in-depth feedback. Such guidance requires a huge pool of exercises, a way to determine student mastery, a way to determine each student's unique problem areas, and specific feedback and exercises for the student at each point of difficulty.

This project will develop software for teaching music students the cognitive skills needed to correctly write and analyze musical exercises in four-part harmony. Drawing upon techniques from the field of AI, the software will constantly assess student learning, determine mastery levels as they change, pinpoint faulty student approaches, provide exact, customized feedback for remediation, and move the student quickly to the most appropriate unmastered level. A set of instructor options will allow teachers to adapt the curriculum to local needs. The software can be easily disseminated to students internationally through existing channels.

Students will achieve improved abilities in the areas of harmonization, chord spelling, recognition and labeling, visual realization of figured basses, and the ability to correlate visual symbols (musical notation) with aural perception and recognition. Through examination of user data collected by the software, new insights into the process by which students come to understand the harmonic process will be gained. These insights may change the way music theory is taught with a shift away from repetitive activities and a shift toward higher-order musical activities.

Programmatic Background

This work is sponsored by the Department of Education under Fund for the Improvement of Post Secondary Education (FIPSE) contract PA1161A1-1226. The contract began in July 1991 and will end in July 1994, the total award amount is approximately $150K.

The primary contractor is the University of Delaware, Department of Music. These researchers will be supported by others from the university's Center for Teaching Effectiveness and from Indiana University. Illinois State University's Music Department within the College of Fine Arts, and its Office of Research in Arts Technology, will act as beta test sites. The University of Kentucky School of Music is another beta test site. The Principal Investigator is Dr. Fred Hofstetter.

Planned Products

Prototype Written Harmony Tutor.

Prior Work

The precedent for this computer-based solution to music instruction problems has already been set at the University of Delaware. Music students at Delaware have been successfully using GUIDO software as part of the ear-training curriculum for over a decade. This software was originally developed at the University of Delaware for the PLATO system and has since been converted to run on IBM and compatible machines. Music instructors have found the GUIDO lessons to be a valuable instructional resource as well as providing as innovative way to track student progress. The GUIDO courseware is used by music departments in schools, colleges, and conservatories worldwide.

Approach

The work in this project is divided into tasks for each year as follows.

Year 1: July 1991 - June 1992.

The following activities have been completed:

- Creation of the curriculum design specification document.

- Creation of the design specification for the instructor editor.
- Completion of the instructor editor software.
- Creation of the design specification for student application.
- Programming of support modules for student notation, audio performance, and user input.

Year 2: July 1992 - June 1993.

The following activities have been completed:

- Completion of the student software.
- Preliminary student testing with University of Delaware music students.
- Testing of instructor editor.
- Creation of draft versions of instructor and student manuals.

Year 3: July 1993 - June 1994.

The first three activities are in progress:

- Completion of final versions of the instructor and student manuals.
- Creation of evaluation questionnaire.
- Testing of the student application at sites around the country.
- Evaluation of questionnaire data from remote sites.
- Respond to feedback from remote test sites.
- Dissemination agreement negotiated and in place for software.
- Presentation of project outcome at a conference to be selected.

Potential Follow-On

The three fundamental skill areas critical to the academic study of music are written theory, sight-singing, and ear-training. These can be thought of as a triangle of inter-related skills. To become accomplished musicians, students must develop a strong relationship between symbols and an internalized representation of sound. Each point on the triangle contributes. For example, written theory enables students to assimilate pitch and time representation systems and the harmonic underpinnings of the tonal system. Ear-training helps students visualize the musical symbols when music is heard or played; this is a sound→sight relationship. Sight-singing empowers students to produce vocally the correct sounds from written musical symbols; this is a sight→sound relationship. When student can write what they hear and hear what they have written, they have mastered the skills in the triangle.

Technology-based instruction can be provided for each point of the skill triangle. The previously developed GUIDO Ear-Training Lessons have effectively provided one point. Merging the two FIPSE efforts, this and the Computerized Sight-Singing Lessons with Intelligent Feedback project (see Section 13.4.3 on page 172), will enable the development of software to address the remaining two points of the triangle. Researchers propose that measuring improvements in the relationship of sight→sound is a major part of this additional work. Measurement of both perceived improvement and actual achievement would be addressed.

13.4.2 Written Harmony Tutor

The exercise generator is at the core of the Written Harmony Tutor. This generator will produce four-part chorale settings with harmonic analysis within the constraints of common practice harmony. A seven-slice data model is used, as illustrated in Table 5 on page 168. Each question generated includes all seven layers in its data structure. When a question is presented, filtering techniques can be used to present any combination of layers to the student as the stimulus and require the user to supply the other slices. Examples of supported question types include soprano harmonization, roman numeral realization, inner-voice completion and lead-sheet analysis. Custom questions are also supported.:

Table 5. Written Harmony Seven-Slice Data Model

Layer Name	Discussion
Soprano Voice	Pitches representing the upper voice in a four-part choral texture
Alto Voice	Pitches representing an inner voice in a four-part choral texture
Tenor Voice	Pitches representing an inner voice in a four-part choral texture
Bass Voice	Pitches representing the lower voice in a four-part choral texture
Figured-Bass	Intervallic representation of the three upper voices in a four-part choral texture
Roman Numerals	Harmonic analysis of all voices in a four-part choral texture
Lead Sheet	Harmonic analysis of all voices in a four-part choral texture

This approach guarantees that a successful solution is always available to the user should they be unable to generate one. (This approach does not guarantee that an exercise partially completed by the user can be finished for them by the software.) Exercises vary in length from one to eight chords. This flexible length permits instruction in the areas of single chord spelling and voicing, connection of two chords, culminating with exercises of a practical musical length.

Effective delivery of feedback information to the student is crucial if skill improvement is to occur. Feedback in this software can be thought of as one of two types: “vertical” or “horizontal.” Vertical feedback relates to errors in a single chord in a progression. Examples of this type of error include spelling, spacing, voice ranges, and doubling. Horizontal feedback relates to errors in the connection of two adjacent chords in a progression. Examples here include parallel fifths and octaves, crossed voices, and awkward or large leaps within a single part.

A matrix approach to student access of feedback information will be used where five layers of information are made available to the user. The x-axis in the matrix represents the number of chords in the current exercise with one column per chord. The y-axis represents the layers of feedback information available, with increasingly explicit information concerning the error being revealed as the student moves down through the matrix. A matrix of this type allows the student to view only one type and level of information desired. By clicking with a pointing device in a cell which is the intersection of a chord number and a feedback information level, the student receives feedback information concerning an error detected. To successively unveil the five feedback layers, a "feedback grammar" which specifies the construction of phrases, clauses, and sentences is used.

Additionally, three types of audio performance are available to the student:

- Exercise Performance. At any point during the answer process, users may request a performance of their exercise. Notes not yet added to the exercise are replaced by rests. An on-screen marker is used to indicate the current "play" location in the exercise. Due to the large number of potentially "correct" solutions, no audio access to the machine-generated correct answer is provided as it may bear little relationship to the solution currently under construction by the user.
- Voice Line Performance. At any point during the exercise creation process, the user may select desired voices for performance. A maximum of three voices may be selected. Any "missing" notes in selected voices are replaced by rests. An on-screen marker is used to indicate the current "play" location in the exercise.
- Chords Performance. Pending the design of a suitable selection mechanisms, individual chords from the current exercise may be performed. This functionality is related to "Happy Chords" in the GUIDO Ear-Training software.

The "unit" is the basic building block of courses. Courses may contain a minimum of 1 and a maximum of 64 units. Instructors create units by means of a specialized editor. Examples of unit parameters under instructor control include the following: competency criteria, judging criteria, question length, question type, and answer timing. (For some unit parameters, the user may be allowed to change the parameter value at run time.) Instructors also have control over rule base parameters concerning triad doubling and voice ranges.

The Written Harmony Editor validates course integrity before courses are made available for student use.

No limit, other than storage capacity, is placed on the number of courses created by an instructor. Only one course may be active at a time; each has a unique signature that is included in student records and used to maintain a strict correspondence between a course and the records of students using that course.

A sample screen from the Written Harmony Tutor is shown in Figure 39 on page 171.

Development Status Software product under development, intended for dissemination for educational purposes.

Architecture The tutor comprises two interdependent programs, the instructor editor and the student software. The user interface is discussed separately.

Instructor Editor The editor allows instructors to create a series of instructional units by making choices in a series of parameters. Instructors have control over such parameters as competency standard, answer time limit, pitch input method, harmonic labels, exercise length, keys to be used, and harmonic content. Instructional unit parameters are verified for consistency when the instructor tries to convert them into a "module" and any elements that need to be amended or added are listed for the user. A verified module file is ready to be used with the student software.

Student Software The Student Software uses the information provided by the instructor (via the Instructor Editor) to present a series of instructional units to the student. Each unit presents a series of harmonic exercises to be solved. Multi-layer feedback is provided for errors detected. When the desired competency is reached, the unit is deemed to have been completed by the student and the judgment marked in the student's records.

User Interface The user provides input using either notational input (via a graphic keyboard for direct manipulation of notational symbols) or by analysis symbol input (via roman numeral harmonic analysis symbols or leadsheet harmonic representation symbols.

Evaluation Status The general applicability of the software will be evaluated by an external consultant. In addition to assessing the strengths and weaknesses of the software, this expert will examine the validity of the rule base used in the lessons, looking for clarity in the music notation, and checking the lessons for ease of use and for effectiveness of feedback.

Functionally, the software developed for the project initially will be tested with music students at the University of Delaware. The developers will work closely with the University's Center for Teaching Effectiveness during the on-campus phase to implement the following evaluation process:

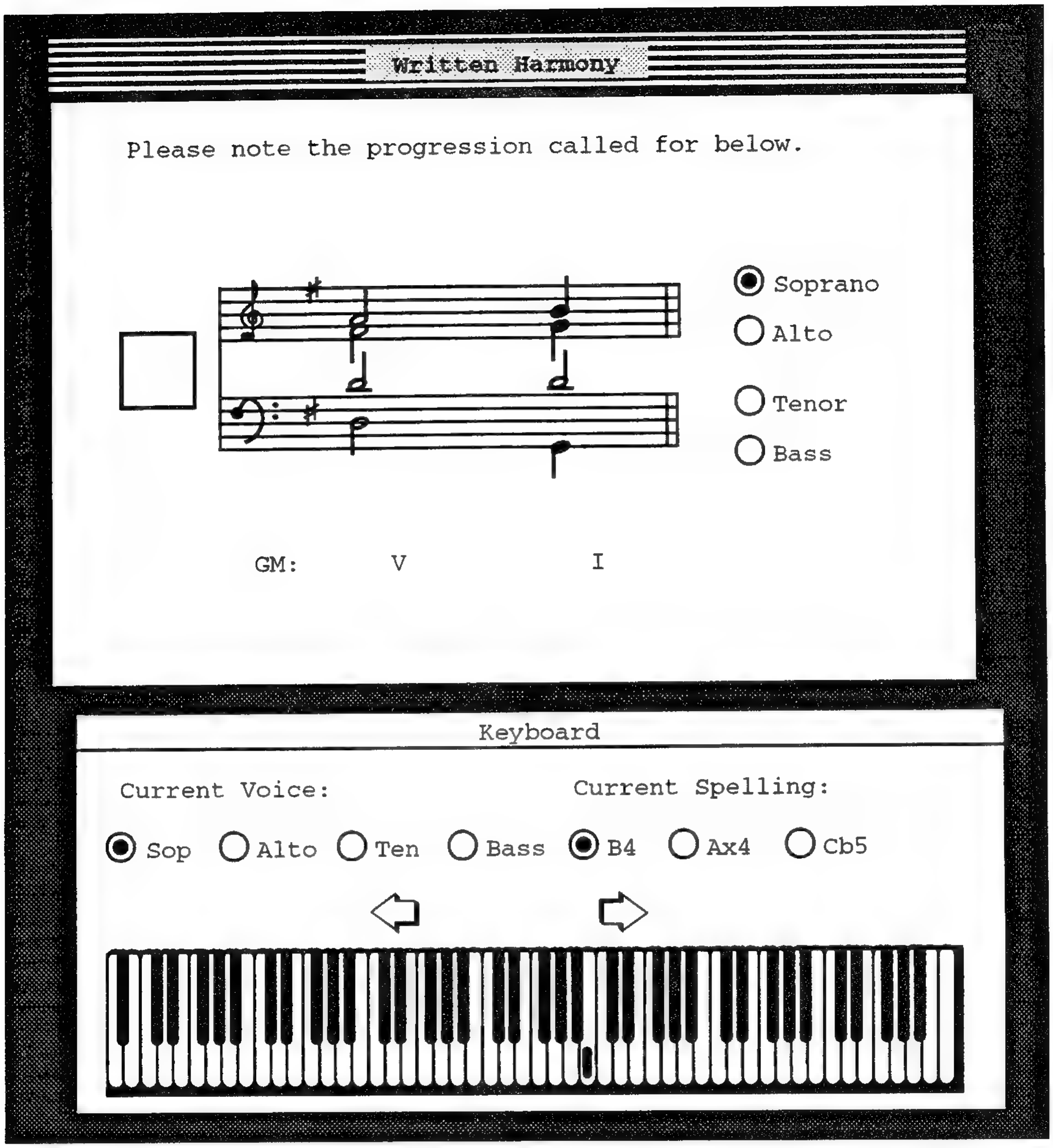


Figure 39. Sample Screen from the Written Harmony Tutor

Formative Evaluation During development, project personnel will gather data by several methods to ensure the lessons meet stated criteria. Student interaction with the lessons will be monitored either by a human evaluator or by video camera. Data gathered by routines incorporated into the code will be analyzed to determine whether particular portions of the lessons are troublesome for students. Questionnaires will be available for students to comment on problems encountered when using the lessons.

Summative Evaluation Because the number of students using these lessons at any time will be small, randomized control group statistical models are impractical. Instead, the developers propose to use interrupted time series analysis to judge the efficacy of the lessons. Data, in the form of attitudinal questionnaires and objective achievement tests will be gathered both before and after the lessons are in use by the students. Demographic analysis will control for any variation in the student population over the years. The attitudinal questionnaires will cover such questions as reaction to the lessons, estimates of time spent on class homework, anxiety level and self-confidence measurements, and a student self-estimate of progress made in the course. An objective performance evaluation instrument will be developed so progress in student achievement during the semester can be reliably measured.

Following the completion of this initial test phase, the software will be tested away from the University of Delaware campus at the beta test sites. In addition to using a questionnaire to measure the efficiency and effectiveness of the software to off-campus users, an attempt will be made to determine how success on tasks required in the lessons correlates with student success on composition and analysis projects in other courses.

Operating Environment The tutor is available for Apple Macintosh and IBM-compatible systems using Microsoft Windows 3.1 or higher. It is written in the C programming language, and was developed using the Extensible Virtual Toolkit (XVT) from XVT Software, Inc.

The Macintosh version provides MIDI (Musical Instrument Digital Interface) format, a standard music industry protocol for computer and electronic instrument communication support via the Apple MIDI Managers. Macintosh built-in audio is supported by means of the toolbox Sound Manager. This software supports devices from AdLib, Creative Labs (SoundBlaster), IBM, Music Quest, and Roland, as well as many other vendors.

13.4.3 Computerized Sight-Singing Lessons with Intelligent Feedback Project

Sight-singing is one of the most important elementary music skills and one that is fundamental to general music understanding and performance. Effective sight-singing practice requires (1) a huge pool of exercises, (2) a way to determine student mastery, (3) a way to determine each student's unique problem areas, (4) specific feedback and exercises for the student at each point of difficulty, and (5) a way to assess whether the instructional approach is working. Ideally, each student would have a private tutor. However, sight-sing-

ing is almost always taught in a classroom with students singing a group, where individual strengths and weaknesses are hidden.

Now that voice-input ("pitch-detection") devices are available for musically equipped computers, software can be written that fulfills the above requirements. The envisioned program will create an inexhaustible supply of exercises and patiently guide the student at each level of mastery. This individualized design will provide the key to a learner-centered educational approach that delivers meaningful and motivating study. AI techniques will be used to analyze performance data and to provide feedback to the students. The software also will collect data for the instructor to use as a basis for ongoing improvement of teaching. A set of instructor options will allow teachers to adapt the curriculum to local needs.

The primary objective is to develop a series of computer-based lessons to supplement traditional teaching in sight-singing courses required of music majors at the University of Delaware. The lessons should enable these students to develop sight-singing skills: (1) more quickly and efficiently, (2) without the constant need of instructor expertise, and (3) in an atmosphere less threatening than the traditional large class setting. Researchers will use formative and summative evaluations techniques to insure that these criteria are met.

However, the software developed during the course of this project will have implications beyond the local level. Music students at other universities, high schools, and other music preparatory institutions, pre- and in-service teachers wishing to improve their singing skills, and those outside a formal instructional environment with similar desires will be able to take advantage of this project. The proposed lessons also can be used with any musical instrument to improve pitch matching, maintaining tonal reference, and pitch accuracy (intonation) skills.

Long-term objectives will focus on two areas: (1) adaptation of the lessons to related curricula, and (2) improvement of instruction through research. First, the lessons will be made available to other educational institutions, both secondary and post-secondary, who have a similar need in sight-singing instruction. In addition, the lessons developed during this project are designed to be easily adapted for use on other vocal and instrumental educational curricula. Second, the data collection routines incorporated into the lessons will enable researchers in sight-singing instruction methodologies, vocal techniques, music perception, and related music fields to gather large quantities of data for their work. The

project researchers will form a network of schools that will share data collected from the completed courseware.

Programmatic Background

This effort is funded under FIPSE contract PA1161A1-1811. It started in July 1991 and is due to complete in June 1995. Total funding of approximately $135K has been awarded.

The University of Delaware in the primary contractor, with researchers from the Music Department, Instructional Technology Center, and the Center for Teaching Excellence all supporting lesson development. The Principal Investigator is Dr. Jon Conrad. Researchers from Cornell and Carnegie Mellon University are acting as expert consultants. Additionally, researchers will be working with music schools throughout the country, including the Peabody Conservatory, in Baltimore, Maryland, and the University of Georgia, to evaluate the lessons. These schools will participate in both formative and summative evaluation of the materials.

Planned Products

Prototype Sight-Singing Tutor.

Prior Work

See Section 13.4.1 on page 165, Prior Work.

Approach

The work in this project is divided into the following tasks.

Year 1: July 1991 - June 1992.

The following activities have been completed:

- Creation of the curriculum design specification document.
- Creation of the design specification for the instructor editor.
- Completion of the instructor editor software.
- Creation of the design specification for student application.
- Programming of support modules for music notation, pitch-detection routines, and user input.

Year 2: July 1992 - June 1993.

The following activities have been completed:

- Completion of the student software.
- Preliminary student testing with University of Delaware music students.
- Testing of instructor editor.
- Creation of draft versions of instructor and student manuals.

Year 3: July 1993 - June 1994.

The first four activities are in progress:

- Completion of final versions of the instructor and student manuals.
- Creation of evaluation questionnaire.
- Testing of the student application at sites around the country.
- Evaluation of questionnaire data from remote sites.
- Respond to feedback from remote test sites.
- Dissemination agreement negotiated and in place for software.
- Presentation of project outcome at a conference to be selected.

Potential Follow-On See Section 13.4.1 on page 165.

13.4.4 Sight-Singing Tutor

There are four principal areas of concern in the design of the sight-singing lessons. The first is that computer interaction with the student be simple, reliable, and helpful. Second, assessment of student performance should be as natural as possible, imitating that of a human instructor. Third, the lessons should allow instructors to track individual student progress and to modify the lessons to fit individual circumstances. Finally, data should be collected so that these lessons and sight-singing instruction in general can be improved.

The computer screen displays a problem (an interval name, or a series of one or more pitches on a staff), a synthesizer plays a reference pitch, and the student sings or plays the notes required into a pitch-detection device. The pitch-detection device converts the student performance to numbers that can be analyzed by the computer using AI techniques and compared to the notes requested by the exercise. Problems in timing and pitch accuracy will be detected and analyzed to determine whether they lie outside the parameters established for the current student. The results of this analysis are then returned to the routines that interact with the student and are used to determine proper student feedback and remediation. On-line help is available at all times. An *Answer* button is available in the *Help* section of the menu bar so that students can choose to hear the correct solution to the problem and avoid becoming unduly frustrated. An example screen display is shown in Figure 40 on page 176.

Great effort has been made to develop the lessons so that student performance is judged by human standards. Lesson developers worked with experts in pitch perception and vocal performance to ensure natural, non-restrictive evaluation.

Lessons are organized as a series of units of increasing difficulty. Using a learner-centered approach, progress depends on the individual student's skills and aptitude. The completion of each unit is recorded for the student and instructor, along with data on any problem areas. The lessons will also provide the instructor with reports on student progress and problems. With the information, the instructor can adjust the classroom and personal instruction accordingly. In this way, the results of the program are tied directly to improvement in the educational process. For example, instructors can change the types of exercises given, the criteria for mastery, the time limit, and other parameters. Based upon these parameters, the program generates the specific exercises; the instructor need not enter long lists of individual musical examples. The evaluation portion of the lessons also is available

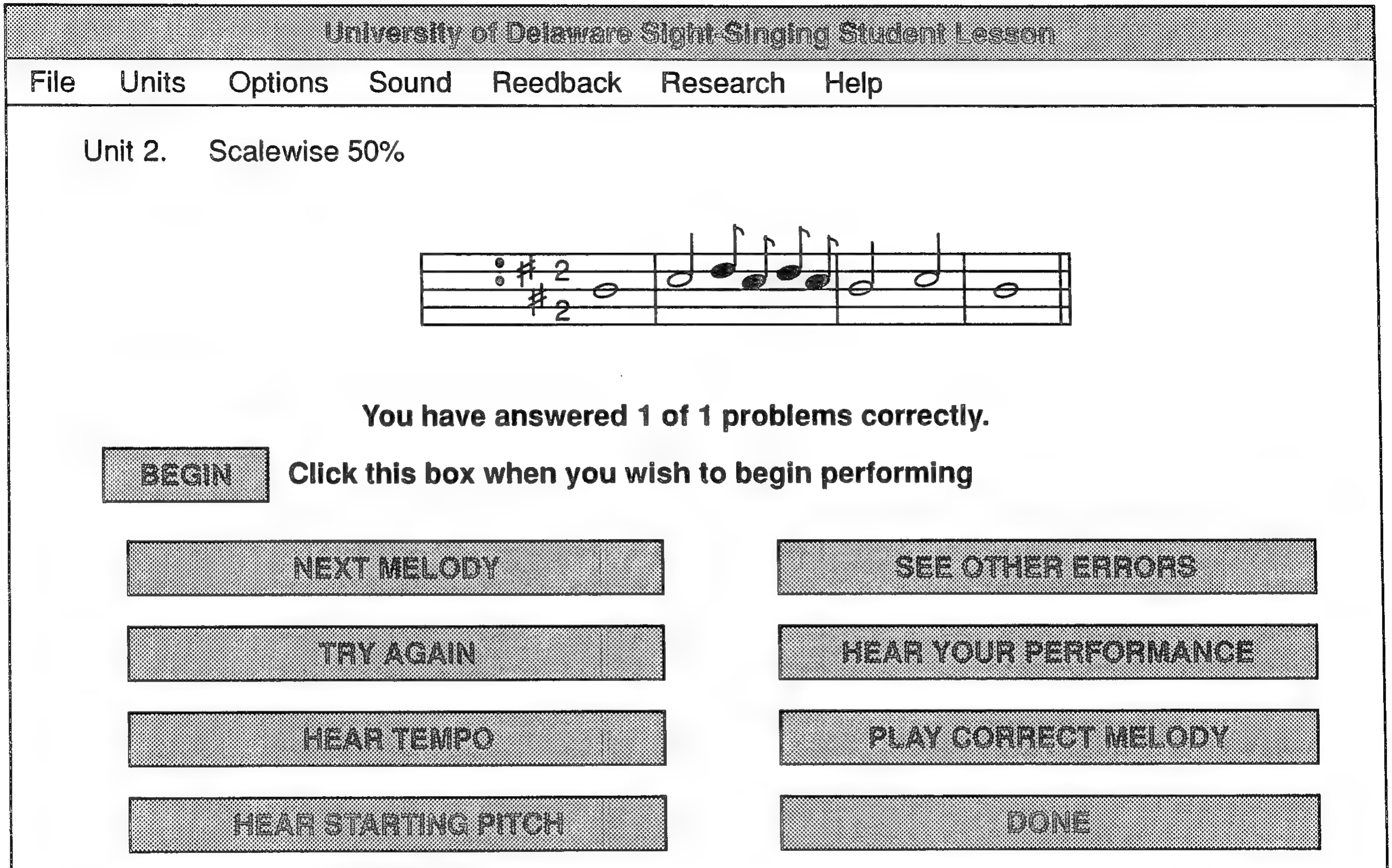


Figure 40. Sample Screen from the Sightsinging Tutor

in an off-line mode. That is, the instructor or student can record a sight-singing lesson on an audio tape and subsequently play the tape into the pitch-detection microphone. The software package will then be able to analyze the performance and given evaluative feedback.

Development Status Software product under development, intended for dissemination for educational purposes.

Architecture The tutor comprises two interdependent programs: the Instructor Editor and the Student Software. The user interface is discussed separately.

Instructor Editor The Instructor Editor allows instructors to create series of instructional units by making choices in a series of elements such as clef, intonation tolerance, competency requirements. A choice of basic content for the unit (pitch matching, interval, chord, scale, melody with or without rhythm) makes available specific control over the precise intervals (or chords or scales or melodic elements) to be presented to the student. The choices are verified for consistency when the user tries to convert them into a "module" and the elements that need to be amended or added are listed for the user. A completed and verified module file is ready to be used with the student software.

Student Software The Student Software uses the information provided by the instructor (via the Instructor Editor) to present a series of instructional units to the student user. Each unit presents a series of musical structures of a particular type (for example, chords, scales, melodies) to the student, to be realized in sound, by singing into the pitch-detection device. Feedback on specific errors is provided, including graphic feedback on desired pitches and actual pitches performed. The most common basis for determining competency in a unit is the correct completion of a certain number of problems out of those most recently generated (for example, four out of the last five problems). When the desired competency is reached, the unit is deemed to have been completed by the student and the judgment marked in the student's records.

User Interface Lessons have an easy-to-use point-and-click interface so students can concentrate on the subject matter rather than on the mechanics of the software.

Evaluation Status This tutor will be evaluated in the same manner as the Written Harmony Tutor (see Section 13.4.2 on page 168), with the following additional actions:

Formative Evaluation At intervals, audio tapes will be made of students using the lessons. An objective expert will evaluate the recorded student performance. This evaluations will be compared to the evaluations provided by the software routines, and these routines will be adjusted to match human evaluation criteria.

Summative Evaluation Short-term progress will be measured by protocol analysis. Audio recordings will be made of students using the computer-based sight-singing lessons. Recordings also will be made of students working as a control group on the same lessons, but without the assistance of the computer. The audio tapes will be edited to remove any clues about the practice environment. Experts will then evaluate the progress of the two student groups.

Testing in Participating Schools

Participating schools will be given the same sight-singing test and attitudinal survey at the end of the Spring semester. These tests will be compared to each other and to those given at the University of Delaware in order to assess substantial differences between the schools. Data from these tests will also serve as controls for future testing in the participating schools.

Operating Environment

The tutor runs on Apple Macintosh and IBM-compatible systems using Windows 3.0. It employs a Pitchrider 400 from IVL Technologies, Victoria, Canada, which itself uses a standard microphone input and produces output in MIDI. The software is written in C and was developed using the Extensible Virtual Toolkit.

14. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

14.1 Mission and Role of Johnson Space Center, Software Technology Branch

The Software Technology Branch (formerly the Artificial Intelligence Section) at the National Aeronautical Space Administration's (NASA) Lyndon B. Johnson Space Center (JSC) is primarily responsible for evaluating and developing new software technology and transferring that technology into NASA's operational environment. To that end, the branch has explored available commercial and research software tools and facilitated their integration into operations at JSC, other NASA centers, and external Government organizations. Where available tools have not answered NASA's needs, the branch has developed its own software tools.

One of the most significant of the branch's activities integrates AI technology with training and tutoring methodologies to address the training of astronauts, flight controllers, and other ground support personnel. NASA's training requirements are gargantuan, and its ability to deliver effective training is continually challenged through rapid personnel turnover, frequent introduction of new technologies and systems, and pressures to reduce costs. The Software Technology Branch (STB) has assumed a leadership role in the development of Intelligent Computer-Aided Training and Tutoring (ICAT) systems. These systems serve to capture the knowledge of the NASA's best engineers and trainers and can provide sophisticated, individualized training to large numbers of personnel in diverse locations.

14.2 Summary of Past ITS-Related work

The branch's ICAT work owes its success to the combined efforts and talents of a number of people. In the summer of 1986, researchers from the University of Houston-Downtown and Software Technology Branch staff members jointly conceived an approach to developing training systems that integrated the previous work of academic researchers with the technology and philosophy of the Software Technology Branch to design NASA's first ICAT system. Subsequently, working with Mission Control Center experts, they produced the prototype of an ICAT system for use by Space Shuttle flight controllers. This ICAT system was designed to train flight controllers in carrying out the procedures neces-

sary in deploying a satellite equipped with a Payload-Assist Module (PAM). The Payload-Assist Module Deploys/Intelligent Computer-Aided Training System (PD/ICAT) evolved from this prototype.

Since that original work, the branch's ICAT team has developed a general architecture for ICAT systems (see Figure 41 on page 181)[1] and built specific applications:

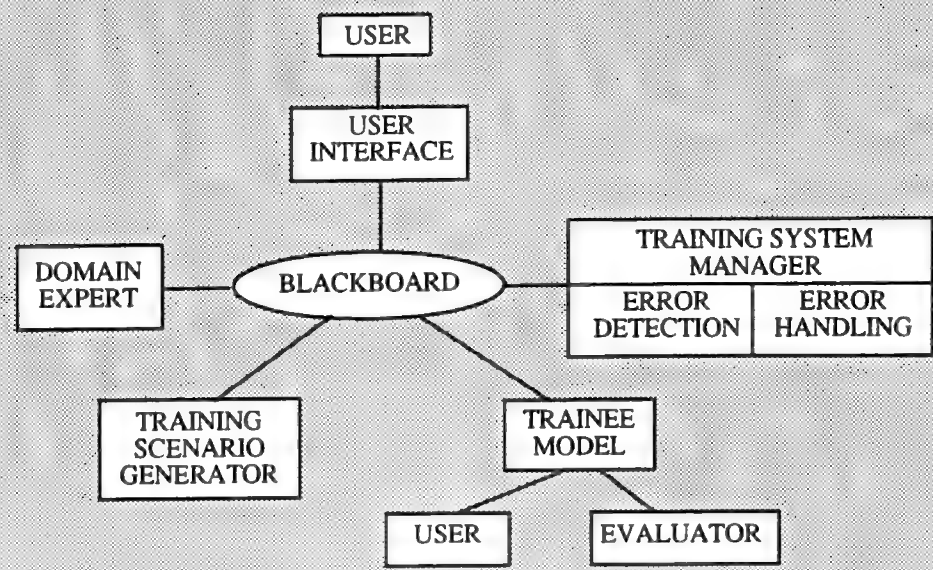
- Instrument Pointing System (IPS)/ICAT for NASA's Marshall Space Flight Center to employ in training astronauts in the operation of the IPS on Astro Spacelab.
- Main Propulsion Pneumatics (MPP)/ICAT for NASA's Kennedy Space Center to employ in training test engineers in testing and troubleshooting the Space Shuttle's MPP system.
- Center Information Systems Computer Operations (CISCO)/ICAT for training computer operators in Johnson Space Center in the operation of the on-site mainframe computer system.
- Active Thermal Control System (ATCS)/ICAT for training flight controllers and astronauts in the operation of the Space Station ATCS.
- SpaceHab Intelligent Familiarization Trainer (SHIFT) for training astronauts and ground-based personnel in nominal and non-nominal operations of the SpaceHab module and associated Space Shuttle systems. (SHIFT was developed in place of the usual Single System Trainer at the Johnson Space Center. It was used in training the crews for STS-57 and is currently in use for STS-63.)

Perhaps equally as valuable as ICAT applications developed for use within NASA is its adaptation for use in education and the transfer of this technology to Department of Defense agencies. From 1988 to 1991, researchers developed the Intelligent Physics Tutor in order to assist high school and college students in acquiring physics problem solving skills. In progress are projects to address adult literacy instruction and the acquisition of early language skills by young children that are prevented by disabilities from the type of peer interaction that serves to build language skills. Projects are now underway to build ICAT applications for the US Army and the National Guard.

Through the development of a general architecture for ICAT systems, software tools to aid in application building, and the transfer of this technology into government,

---

[1] This architecture is the subject of U.S. Patent Number 5,333,422 awarded to NASA.



NASA/JSC projects have served as vehicles to aid in the design and refinement of an architecture for ICAT that has significant domain-independent elements and is generally applicable to training in procedural tasks.

The system architecture is modular and consists of five basic components:

- A user interface that permits the trainee to access the same information available in the task environment and serves as a means for him to take actions and communicate with the ITS.
- A domain expert that can carry out the task using the same information that is available to the trainee and that also contains a list of "mal-rules" (explicitly defined errors that novice trainees commonly make).
- A training session manager that examines the actions taken by the domain expert (of both correct and incorrect actions in a particular context) and by the trainee and takes appropriate action(s).
- A trainee model that contains a history of the individual trainee's interactions with the system, together with summary evaluative data.
- A training scenario generator that designs increasingly complex training exercises based on the knowledge of the domain expert, the current skill level contained in the trainee's model, and any weaknesses or deficiencies that the trainee has exhibited in previous interactions.

Provision is made for the user to interact with the system in two distinct ways and a supervisor may also query the system for evaluative data on each trainee. The blackboard serves as a common repository of facts for all five system components. With the exception of the trainee model, each component makes assertions to the blackboard, and the expert system components look to the blackboard for facts against which their rule patterns match.

This architecture was originally implemented in a Symbolics 3600 Lisp environment using Inference Corporation's ART for the rule-based components. The architecture is currently available for Unix workstations. The user interface is implemented in X-Windows, the rule-based components in CLIPS, and supporting code in C.

Figure 41. Overview of NASA's ICAT Architecture

industry, and education, ICAT has made and will continue to make important contributions to cost reduction, increased training throughout, and improved human performance.

14.3 Overall ITS R&D Program

The STB R&D program consists of (1) application development to address specific agency needs; (2) the development, refinement, and evolution of a general architecture for ICAT systems that enables the rapid creation and routine maintenance of specific applications; (3) the development of software tools that, in conjunction with the general ICAT architecture, support the creation and maintenance of ICAT applications by training personnel with limited programming experience; (4) the transfer of this technology to other government agencies, industry, and academia; (5) the integration of ICAT technology with other emerging technologies (for example, virtual reality) to enhance its utility, range of applicability, and portability; (6) the investigation of ICAT application effectiveness in training transfer; and (7) the continuing investigation of new technologies that can enhance training, education, and related activities.

14.4 On-Going ITS-Related Tasks

Current activities are supported by a Technology Development Contract, NAS 9-18630, held by LinCom Corporation to provide contractor support to the STB. LinCom is the prime contractor and the contractor team includes I-NET and SAIC.

A grant to the University of Houston-Downtown, NAG 9-713, supports Dr. Hugh Bowen Loftin and his students. Dr. Loftin provides technical guidance to the contractor and civil service Training Technologies Team and carries out research on training effectiveness and training systems design.

From time to time, the STB supports SBIR grants in the areas of ICAT and virtual reality (as it relates to training). Other grants to universities support short-term research with specific foci.

14.4.1 Conduct of Fire Trainer Automation Project

Gowen Field, Idaho, is one of the key sites for armor crew training and houses state-of-the-art training devices and simulators. Currently, Conduct of Fire Training (COFT) simulators provide M-series and Bradley tank gunnery training to the US Army, Reserve, and National Guard troops. The current implementation of the COFT requires one trainer for each system when employed as a Mobile COFT (M-COFT) or as an Individual COFT (I-COFT). This need for one trainer for each system fixes training costs and can result in

under use of the COFT system. Moreover, the variable experience and backgrounds of the trainers dictate that some nonuniformity of training exists and that verification of training is difficult, if not impossible, to obtain.

ICAT systems capture the knowledge of the best trainers and subject matter experts and encapsulate this knowledge in a form that provides each trainee with his "personal" trainer. Such systems can be linked to existing simulators (such as the COFT) and provide scenario selection, real-time assistance, and after-action reviews for the trainees. In addition, since the ICAT system builds and maintains a unique student model for each trainee, it can optimally select training scenarios that can serve to both remediate observed trainee weaknesses and also to advance the trainee's experience level toward the ultimate training goals. These systems observe trainee actions as they are performed and can provide "just in time" coaching, much as a human trainer can provide as he "looks over the shoulder" of the trainee. Thorough after-action reviews are provided to the trainee (in hardcopy form if desired) and data on individual trainees or groups of trainees can be provided to training and command personnel. Finally, once an ICAT system is developed and verified, it can be replicated with no additional cost (excepting needed hardware) and can deliver uniform and verified training to personnel at all locations.

The objectives of this project are as follows:

- Automation of "instructor" functions of an Instructor/Operator within the COFT facility (fixed or mobile) and integration, through external links, with the existing COFT simulation. The three main instructor functions of an Instructor/Operator to be implemented are exercise selection/simulator initialization, performance monitoring/coaching, and after action-review.
- Development of an automated tank commander and tank gunner system and its integration with the "Automated Instructor/Operator" or COFT/ICAT developed through the previous objective.

Programmatic Background

This project began in July 1993 and is due to complete in July 1994. It is funded by ARPA, with the amount of $360K. Additional support has been provided to Infotech to support ICAT integration with the existing M-COFT system.

NASA/JSC research staff are supported by researchers from the University of Houston-Downtown and LinCom. The Principal Investigator is Dr. Hugh Bowen Loftin from the University of Houston-Downtown.

Planned Products

COFT/ICAT prototype for the M1A1 COFT.

Previous Work

Previous ICATs employing the same architecture include PD/ICAT, MPP/ICAT, CISCO/ICAT (see Section 14.2 on page 179).

Approach A three-phase approach is being taken for this project.

Phase 1: NASA/JSC will use its general ICAT architecture and ancillary software tools to develop an ICAT system that automates the instructional role of the COFT Instructor/Operator. Specifically, this ICAT system (COFT/ICAT) will build a model of each trainee using a given system. This model will be used to select an appropriate scenario for each training session. Data reporting trainee actions will be used to modify the student model, remedy identifiable trainee errors, and generate an after-action review for use by both the trainee and training personnel. Summary data on groups of trainees will also be provided for use by training personnel. During this phase of the project, the COFT/ICAT system will be hosted on a Unix platform.

In addition to all existing documentation on the COFT training procedures, access to the best COFT trainers as well as field personnel responsible for training in National Guard units must be provided to enable the collection of uncodified experience. The heuristics used by the best trainers can thus be captured and incorporated into the COFT/ICAT system. Access to those contractors and personnel that support and maintain the actual COFT software and hardware is also required.

Phase 2: The ICAT system will communicate with existing COFT systems via standard serial data links. Given that the current COFT simulation runs under VMS on a DEC platform, analysis will be performed on the COFT simulation design and architecture. Researchers will need documentation and source code to enable the effective linking of the simulation with the ICAT architecture. The primary issues to be addressed are (1) the ICAT system's ability to initialize a selected training scenario, and (2) the transmission of trainee actions to the ICAT system.

Phase 3: The third phase will address the development of an automated tank commander and gunner system and its integration with the COFT/ICAT system developed in Phase 2. This system will be rehosted to the computer platform used in Phase 2 to run concurrently with the COFT/ICAT.

Potential Follow-On Work ICAT capabilities and system integration mechanisms developed for this project can be adapted for other combat vehicle simulators used by the Army. For instance, ARPA is interested in the integration of an ICAT system with the Bradley COFT after the current project. Other potentials for the ICAT technology in the Army training environment include development of a stand-alone, tabletop system hosting both the ICAT and the simulator on a single computer platform.

14.4.2 COFT/ICAT

The COFT/ICAT system is an integration of a workstation-based ICAT system and the existing COFT simulator hosted on a VAX system. Data communication between the two systems will be accomplished through an Ethernet connection. The ICAT will automate the Instructor/Operator's role in the COFT facility. For the instructional role, the ICAT is capable of (1) selecting exercises, (2) providing real-time coaching by digitized sound, and (3) reviewing performance of the crew after an exercise. For the operator's role, the ICAT will simulate the tank driver and loader's functionality normally performed by the Instruc-

tor/Operator. In the absence of one of the two crewmembers, either the tank commander or the gunner, the ICAT will be capable of simulating actions normally performed by the absent crewmember. Speech recognition technology will be used to capture voice interaction and commands from the crew.

Development Status	A demonstration of just the ICAT functionality was completed in November 1993. The demonstration prototype was a stand-alone system hosted on a Sun Sparc 10 workstation. It consisted of an expert system encapsulating the instructional knowledge of an Instructor/Operator and a user interface simulating the COFT environment. An integrated prototype of the ICAT and COFT simulator is currently under development.
Architecture	The COFT/ICAT is based on the standard NASA ICAT architecture summarized in Section 14.2 on page 179.
Evaluation Status	The integrated COFT/ICAT will be delivered to the Idaho National Guard. There are no formalized evaluation plans at this time. However, feedback from users in Idaho will be incorporated into the upcoming Bradley COFT/ICAT project.
Operating Environment	The COFT/ICAT will have to interface with the current M1A1 Mobile COFT hardware/software configuration in Idaho. It is assumed at this time that the COFT host computer cannot support the added load of the ICAT system and/or the Automated Tank Commander/Gunner system. Thus, a Unix system will be required to interface to the COFT simulation and to host the software developed by this project.

14.4.3 Early Language Intervention Project

Of approximately 20 million people in the United States with some form of disability, more than 4 million are infants, children, or youths. Recent legislation from the Federal Government puts an increasing emphasis on the early education of children with disabilities. The Individuals with Disabilities Education Act (IDEA) provides for early intervention in the education of children with disabilities from the age of three. Programs such as Head Start and state-mandated programs for early intervention and education of children at risk for developing disabilities may serve children from the age of three months.

A primary focus of early intervention programs and special education programs for young children is the development of basic language and communication skills. Without a firm understanding of basic concepts, children will be unable to succeed in school programs. Education research has indicated that the use of computers with speech output to teach emerging language skills can significantly increase the language skills of young children with disabilities.

This project has the following objectives:

- Recast the student model, global strategy model, and local strategy modules of the Intelligent Physics Tutor (IPT), see Section 14.4.7 on page 190) to address early language acquisition by children who can benefit from early intervention.
- Design and develop an appropriate user interface for this domain that is suitable for use by its intended audience.
- Design and implement suitable screening mechanisms to identify and characterize the language skills of initial users of the system.
- Implement a Teacher's Window (from the Intelligent Physics Tutor) application that will use the Early Language Intervention System (ELIS) student model to provide guidance to teachers in the selection and use of ancillary instructional aids.
- Transfer technology from NASA to the private sector.
- Develop and market one or more software products for use in educational settings.

Programmatic Background This project began in 1992 and is due to complete in 1995. NASA has provided $290K in funding.

Researchers at the STB will be supported by researchers from the University of Houston-Downtown, the Oregon Department of Education, and through a subcontract with Laureate Learning Systems. The Principal Investigator is Dr. Hugh Bowen Loftin from the University of Houston-Downtown.

Planned Products Productized ELIS.

Previous Work The ELIS system inherits its approach from the Intelligent Physics Tutor and, to a lesser extent, from the IPS/ICAT and SHIFT.

Approach A three-phase approach is being taken for this project.

Phase 1: A description of the ELIS lesson design, screening methods, and interface design will be prepared. The Intelligent Physics Tutor architecture will be adapted to provide graphical knowledge editing tools. These tools will allow Laureate's curriculum designers to script ICAT lessons without programming. Based on the results obtained by testing an initial prototype ELIS, a final prototype capable of supporting one word to two-word transitions will be developed.

Phase 2: Formal field testing of the ELIS prototype will be conducted. Meanwhile, researchers will develop and integrate new technologies in speech recognition and advanced multimedia. Speech analysis and recognition will allow more effective diagnosis of each student's language development level, while advanced multimedia techniques will allow new types of video-based remediation and instruction. The first, productized version of ELIS will be developed and delivered.

Phase 3:	The first system will be expanded to address emerging language skills of children up to five years of age. A fuzzy logic student model and instructional strategies will be integrated into the system. Virtual environment technologies will also be integrated into the system.
Potential Follow-On Work	On the completion of this project, Laureate Learning Systems will continue developing new products based on this technology. These products will likely expand the curriculum to support advanced sentence formation.

14.4.4 ELIS ITS

ELIS will provide vocabulary and initial grammatical instructional for the lexical acquisition period of late one word to beginning word combination stages (that is, the ages of 13 or 14 months to 24 months), including SVO (subject-verb-object) parameters. The system will also address the two-word phrase in expanding the lexical categories of noun, verb, adjective, and preposition into full phrases.

The ELIS system will be used in both special education and traditional elementary school classrooms. The system displays pictures of locations containing common objects, then verbally questions the student about the name and function of each object. The system automatically evaluates and maintains records of each student's development level. It uses this information to script custom sequences of objects appropriate to each student. This allows each student to have one-on-one attention not normally available in a public school setting.

Development Status	Prototype and production systems under development.
Architecture	The ELIS ITS is based in the standard NASA ICAT architecture summarized in Section 14.4.2 on page 184.
Evaluation Status	The ELIS system will be evaluated under the direction of Ms. Gayle Bowser of the Oregon Department of Education, in cooperation with linguistic experts from Laureate Learning Systems.
Operating Environment	Think C and CLIPS will be used for basic code development. Tools such as TARGET may be used for knowledge acquisition and to support code development. Macintosh Toolbox or XVT will be used to support interface development. Both the development and delivery platforms will be Macintosh LC II.

14.4.5 Horizontal Integration of Battlefield Augmentation Project

The US Army and DOD have a need to conduct experimental testing and analysis in support of new hardware requirements development and training development. In particular, the Mounted Warfare Test Bed in Fort Knox has a goal to develop the software and

procure the hardware necessary to determine the degree to which new technology can be implemented.

Often times, armor soldiers come into the battle simulation facilities with little current knowledge of the latest version of equipment in their tanks. In an effort to alleviate this problem, this project will address the development of an ICAT for training soldiers on how to use the IVIS communication device. Soldiers will receive one-on-one training from an ICAT which represents the current version of the IVIS equipment, so they will be well prepared before undertaking a full-scale simulated battle. Developers at the STB have already created a system to address similar problems in the NASA/JSC environment and the training needs of the Army can be addressed with this technology.

This project calls for the development of an ICAT and several communication device emulators to correctly model information passing on the battlefield between tanks, helicopters, and armored personnel carriers. These products are the InterVehicular Information System (IVIS), IVIS/ICAT, Digital Message Device (DMD), Forward Entry Device (FED), and Airborne Target Handover System (ATHS). By training with these devices, armor soldiers can not only train on models of the latest equipment in the field, they can also make suggestions as to how they can be improved. These improvements can then be reflected back into the software, tested, and issued as hardware requirements to a vendor.

Programmatic Background

This project began in July 1993 and is expected to be completed in January 1995. The total funding amount is approximately $1.1M.

The Principal Investigator is Dr. Hugh Bowen Loftin from the University of Houston-Downtown. STB researchers will additionally be supported by researchers from LinCom Corp., I-NET Corp. and SAIC.

Planned Products

IVIS, DMD, FED, and ATHS emulators. IVIS/ICAT.

Previous Work

Related work includes development of the CLIPS ITS, IPS, IPT, and SHIFT. It is closely related to the current development of version 2.0 of the general ICAT architecture.

Approach

The work has been structured into four phases. Phases 2 and 3 will be executed concurrently at the completion of Phase 1. Phase 4 will be performed at the completion of Phases 2 and 3.

Phase 1: Emulations of the IVIS, DMD, and ATHS will be developed. Communication between these emulations will be available (assuming Ethernet connectivity exists) and each will replicate the behavior of the appropriate device as described in Army-supplier user's manuals. This will require knowledge acquisition and design activities. Interface prototypes will be subject to customer review. The operability of the systems on customer hardware will be verified, along with network communications between the Tactical Operations Center (TOC) and Combat

Vehicle Command and Control (CVCC) vehicles. In addition to various system, and software information, the Army will provide access to one or more domain experts who fully understand how the existing systems work.

The IVIS emulation will be built to run in the CVCC TOC and on the 8 CVCC-equipped simulators. It will also be installed in the M1A2s if required. The CVCC TOC will be capable of communicating with the IVIS simulators, and the IVIS simulators with the CVCC TOC (again, assuming Ethernet connectivity exists). A DMD emulation will be built to be delivered as a tabletop, workstation-based version compatible with both a 19" and 13" monitor. The ATHS emulation will be delivered as a tabletop, workstation-based version with a 19" monitor.

Phase 2: Researchers will work with Army IVIS instructors in assessing the training requirements. An initial IVIS/ICAT design will be developed and reviewed. A user interface prototype will be developed and evaluated. Also, a mid-term review of developed lessons will be conducted with Army personnel. The IVIS/ICAT will be developed and delivered on a Sun Sparcstation to provide training on a pre-specified version of the IVIS software as determined by the Army. It will be capable of working with a joystick similar to the one in use in the combat vehicles.

A major assumption for this phase is that the Army will provide access to one of the tabletop MacIvory IVIS training systems or to one of the human IVIS trainers capable of working with the ICAT development team to assist in lesson development and in assessing the best techniques for providing the required training. It is also assumed that the Army will supply the version of IVIS software to be modeled.

Phase 3: Phase 3 requires developing the complete emulation of the FED. This will necessitate knowledge acquisition and design activities. Initial interfaces will be subject to customer review. The systems' operability on customer hardware will be verified, along with the network communications between TOC and CVCC vehicles.

Phase 4: This final phase requires preparing the final documentation for the developed systems.

Potential Follow-On Work Future potential work includes the following:

- Upgrades of IVIS/ICAT for compatibility with newer releases of tank hardware.
- ICAT for DMD and ATHS.
- Integration of IVIS/ICAT in the classroom with the battle simulation facilities.
- IVIS/ICAT to train more than one person at a time over a network.

14.4.6 IVIS/ICAT

IVIS/ICAT will be used specifically to train soldiers on IVIS interface operations and battlefield logistics. Students will be provided one-on-one training via the ICAT in a classroom. The IVIS emulators in the classroom will be networked to allow the soldiers to practice sending battlefield messages among themselves.

Development Status The IVIS/ICAT is currently under development and scheduled for completion in December 1984. The prototype version is being built on a Sun workstation.

Architecture The IVIS/ICAT is based on the standard NASA ICAT architecture summarized in Section 14.2 on page 179.

Evaluation Status The IVIS/ICAT will be tested and evaluated in the Armor School at Fort Knox to assess its accuracy and effectiveness.

Operating Environment The IVIS/ICAT is being developed to run on a 486 PC running Unix. Versions will also be available for Sun workstation platforms. In order to use the ICAT in a "simulation only" mode with multiple personnel passing information back and forth, the PCs will require networking which supports TCP/IP.

Future System Development Plans The IVIS/ICAT currently uses Simnet protocol for network communications. It will be updated to use DIS protocol to remain compatible with modifications taking place in the testbeds at Fort Knox.

14.4.7 Assessing the Potential of Virtual Realities in Science Education Project

The goal of this research is to chart the potential opportunities and challenges of VR for learning in science. The researchers will integrate a VR interface with intelligent coaching, distributed simulation, and visualization techniques to provide a Virtual Physics Laboratory. This will be a first step toward laboratories for other sciences, such as chemistry, life sciences, and planetary sciences. Experiments in the Virtual Physics Laboratory will focus on remediating common misconceptions about physics based on learners' misgeneralizations of real-world experiences. Using this laboratory, cognitive trials will be conducted with individual learners to research how various aspects of the virtual environment (immersion, visualization, and contextualized intelligent aids) can be designed to optimize learning.

This project's results will aid in establishing the degree to which VR is a useful adjunct in science education. As moderate-cost, high-performance VR systems emerge, the outcomes of this project will play a crucial role in defining design principles for the educational use of VR technology.

Programmatic Background This project began in February 1994 and is due to complete in January 1996. The total funding amount is approximately $936K.

The STB researchers are supported by researchers from the University of Houston-Downtown and George Mason University. The Principal Investigator is Dr. Hugh Bowen Loftin from the University of Houston-Downtown, and the Co-Principal Investigator is Dr. Christopher Dede from George Mason University.

Planned Products Prototype VPL II. Research results from cognitive trials and the design principles these induce.

Previous Work

In constructing a synthetic environment that integrates intelligence and visualization with VR, the researchers will build upon an existing prototype of a Virtual Physics Laboratory (VPL) and the Intelligent Physics Tutor. The development of VPL I was sponsored by NASA/JSC and the Texas Advanced Technology Program. Figure 42 on page 192 provides an overview of VPL I.

The development of the Intelligent Physics Tutor was funded by NASA, the Texas Advanced Technology Program, Apple Computer's Apple Classroom of Tomorrow, the Brown Foundation, and Pennzoil. Development of the IPT is completed; see Figure 43 on page 193 for an overview of the system. It has been deployed in two high schools and one university, and has been evaluated through a controlled study. This tutor has been licensed to a vendor for commercialization.

Approach

This study centers on exploring the usage of VR technologies in science education. The two-year first phase of this research will address the following:

- Integration of VR with intelligent coaching technology and with visualization techniques.
- Cognitive trials with individual learners to research how various aspects of VR (immersion, visualization, contextualized help) can be designed to optimize learning.

The researchers will build upon the VPL I prototype, adding enhancements that permit the study of collisions in one or two dimensions, rotational motion, and the behavior of a mass-spring systems. The project will explore the extent to which manipulating learners' visual, auditory, and tactile receptors may induce subtle new types of misconceptions about physical phenomena. Researchers will evaluate the degree to which current interaction methods influence the learning process and will determine which interaction metaphors offer the greatest potential to enhance learning of, or, at worst, not affect, students using a virtual environment. Controlled trials with individual subjects will be used to compare learner mastery of physics concepts based on virtual experiences to student outcomes using more traditional approaches.

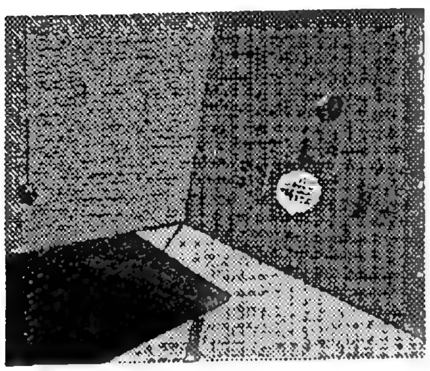
Selective integration of the IPT into the VPL will be explored. In particular, the tutor can be used to pose an appropriate problem to the laboratory user (based on his student model), while the laboratory is used to realize the physical situation described in the problem and to acquire the data needed to solve the problem. The tutor can then provide guidance to the student as data is acquired and analyzed. By conducting cognitive trials with and without the presence of the tutor, the relative leverage for learning that this type of enhancement provides can be determined. Such studies will also aid in developing design principles for intelligent coaching in immersive virtual environments.

Additional work will embed in the virtual learning environment "sensory transducers" that allow users' eyes, ears, and hands to access previously imperceptible phenomena (such as a molecule) and "cognitive transducers" that perform a similar function for intellectual entities. Again, controlled trials will be used to compare student mastery of physics with and without these visualization aids.

Purpose: The Virtual Physics Laboratory (VPL) was designed and built to address many of the well-documented misconceptions that physics students typically carry with them as they enter—and leave—physics courses. This includes misconceptions about key concepts (such as the nature of mass, acceleration, momentum, charge, energy, potential difference, and torque), as well as fundamental principles and models (such as Newton's laws, conservation laws, the atomic model, and electron flow models for circuits).

Hardware & Software Environment:
Hardware consists of helmet-mounted, color stereoscopic displays; 3D acoustic environment; systems for obtaining input from hand gestures; Polhemus magnetic orientation system for tracking observer's viewing position, head position, and hand positions, two (one for each eye) Silicon Graphics 4D/320VGX for rendering. The initial software environment of VPL Research's Swivel 3D and Body Electric is being changed to NASA's Solid System Modeler, Stereo Display Manager, and rendering software.

Virtual Laboratory Environment:
The laboratory is represented as a large room containing a table. The walls, ceiling, and floor are delimiters of the working space. They are also surfaces against which objects in the room can rest or collide. The table represents a workspace on which students can perform experiments; it is also an object which can be relocated in the room. Additional objects include a red ball, blue ball, and a pendulum hanging from a rod mounted on one of the walls; these objects obey physical laws.
Available interactions are via hand gestures, controls, and measurement tools. Controls, located along one wall, include switches for friction and air drag, sliders for manipulating variables such as the coefficient of restitution, and reference objects for gravity control. Measurement tools include a digital display, tracer buttons, distance measurement balls, freeze button, reset button, and overall panel manipulation.



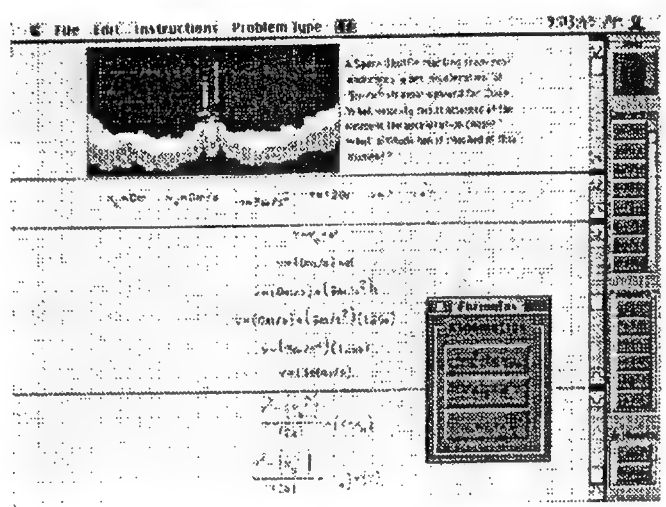
Sample Experiments:

- *Measure the period of pendulum for different lengths and different magnitudes of gravity.* The controls permit the user to measure the period of the pendulum with different lengths (a typical "real" laboratory experiment) and with different gravitational accelerations (not possible in the "real" laboratory). Atmospheric drag can also be reduced to zero or set to a predefined value.
- *Measure the average rate of loss of energy of a ball caused by air drag when dropped from different heights, and under different gravitational accelerations.* Using the trace and freeze facilities, the user can easily record the trajectory of a ball. Varying its coefficient of restitution and atmospheric drag permits measurement of the energy lost on collision with a surface or through interaction with the atmosphere. It is also possible to drop two balls with different coefficients of restitution and observe their subsequent behavior as they bounce repeatedly from the floor.
- *Compare the trajectories (especially range and maximum height) of an object projected in two dimensions with and without atmospheric drag.* The trace facility allows the user to record the entire trajectory of an object, and the measurement facility provides the means to directly measure range and maximum height. Thus, the user can project one of the balls in the room with drag "on" and characterize its trajectory and then repeat the process with drag "off."

Figure 42. Overview of the Prototype Virtual Physics Laboratory

Physics Tutor Goals:

- Develop student problem-solving skills necessary to succeed in physics and other scientific and technical disciplines.
- Provide a tool for student to use on their own time for additional practice without requiring a teacher to be present, but with the same help as if the teacher were present.
- Provide a tool which is effective in tutoring the majority of students, thereby freeing up more of the teacher's time to spend with the very bright and the very weak students.
- Enable teachers to customize the curriculum and lesson plans for their class and be provided with student and class performance data for evaluation.
- Evaluate the use of animation, color video, digitized sound, and speech in tutoring systems.



Physics Tutor Features:

- Provides individualized instruction for each student accommodating the student's pace and ability.
- Robust teacher's window capable of reporting both individual and class performance to the teacher; lesson plans can be customized according to each teacher's preference.
- Detailed student model maintains record of every student action and generates diagnostic messages based on past history.
- Flexible error detection system allows problems to be solved more than one correct way.
- Extensive help facility explains what a student did wrong and provides hints upon student request.
- Designed to be incorporated into a first semester high school physics curriculum.

Physics Tutor Curriculum:

- 1D kinematics
  - -Position and displacement
  - -Average/constant/instant velocity
  - -Uniform acceleration
  - -Acceleration of gravity and free fall
  - -Multi-part problems
  - -Two-object problems (connected by time)
- 2D kinematics
  - Projectile motion (without air resistance)
  - Vector analysis
    - -Vector addition and subtraction in 2D using components
- Statics and dynamics
  - Forces
  - Vector addition and substraction of forces in one or two dimensions (net force)
  - Newton's laws of motion (without friction)
    - -1D dynamics with 1 object
    - -1D dynamics with 2 or more objects
    - -Two-object problems in 2D
      - Inclined plane problems
  - Friction
    - -Normal force
    - -Newton's laws of motion with friction
  - Statics
    - -1D equilibrium
    - -2D equilibrium

Figure 43. Overview of the Intelligent Physics Tutor

In work commissioned by the Army, the Co-Principal Investigator Dr. Dede, is undertaking preliminary explorations of the dimensions of consciousness, meaning, and mood that underlie immersion. Experiments on incorporating factors that affect one's subjective sense of realism into the VPL will be conducted. There will also be an exploration of how low the sensory fidelity of these learning environments can be before the feeling of immersion disappears, and how student outcomes alter at different levels of immersion.

Experiments with transmitting virtual environments over commercial telephone lines will be conducted. This will provide a means to explore shared, cooperative learning experiences across barriers of distance. In addition, these interactive, shared virtual worlds will enable experiments to determine the leverage that collaborative learning provides in immersive artificial realities. The types of functionalities that are important in empowering cooperative learning in such an environment will be investigated.

Potential Follow-On Work

If the outcomes of this project are promising, a second-phase project will be proposed to develop and field-test virtual realities structured to supplement curricular units in physics and chemistry (and, potentially, in planetary sciences and biology) for high school and first-year college students. Additional objectives for such later work include the following:

- Formative evaluation, refinement, and assessment of these units in typical public school settings.
- Generalization of development principles from these paradigmatic models for virtual environment applications in education.

14.4.8 Authoring System Development Project

Although there has been much research into ITS, there are few authoring systems available that support ITS metaphors. Instructional developers are generally obliged to use tools designed for creating on-line books. This goal of this project is to develop an authoring environment derived from NASA's research on ICAT. The ICAT metaphor has proven effective in disciplines from satellite deployment to high school physics. This technique provides a personal trainer (PT) who instructs the student using a simulated work environment (SWE). The PT provides individualized instruction and assistance to each student. Teaching in an SWE allows the student to learn tasks by doing them, rather than by reading about them. Different fidelities of simulated environment and personal trainer are used, depending on the needs of the training.

This authoring environment will expedite ICAT development by providing a tool set that guides the trainer modeling process. Additionally, this environment provides a vehicle for distributing NASA's ICAT technology to the private sector.

Simulating a work environment currently requires special graphical user interface programming skills and often experience in engineering simulations. This project will attempt to integrate a commercial simulation package, allowing an instructional designer to both graphically lay out the SWE and "wire" the resulting simulation together. Likewise, trainer modeling, the process of overlaying an SWE with lessons and providing remediation, is involved and time consuming. This project will develop tools to guide the instructional designer through this process.

Programmatic Background

This project has been in progress since October 1992. The effort became joint with the NASA/JSC Space Flight Training Division in October 1993. Requirements development was completed in January 1994 and development commenced at that time. Completion is due in early 1995.

Total funding in FY94 is approximately \$50K, from external sources, and \$60K from internal sources.

Planned Products

Prototype ICAT authoring environment.

Previous Work

The effort builds on much prior STB work. Incorporating a simulation in an intelligent training system was pioneered by JSC in the PD/ICAT. Both the action evaluator (see Section 14.4.9 on page 197) and its operating procedure language stem from the SHIFT work. The Operating Procedure Language (OPL) was derived from a NASA knowledge acquisition tool, Task Analysis Rule Generation Tool (TARGET), which in turn borrowed a notation from the Navy tool VISTA. (TARGET is an ICAT tool intended to permit experts to visually describe procedural tasks and to be a common medium for knowledge refinements by the expert and knowledge engineer.) The PT concept is a refinement of the IPT's "local strategy." The local strategy's misconception theory was heavily influenced by concepts pioneered in the Buggy ITS.

Approach

Phase I of this project resolves several technical problems that have impeded the development of prior ICAT systems. The tasks in Phase I are as follows.

Task 1: *Identify and Integrate a Commercial Tool That Can Create Both the Graphical User Interface and the Underlying Engineering Model of an SWE.* The first task of this project during Phase I will be to research available simulation building tools, choose the most effective one for building SWEs, and develop the needed code to integrate this tool with the ICAT run-time algorithms.

To judge whether or not a product will be able to be integrated into an ICAT system, the researchers will investigate and answer the following questions:

- Can the product be integrated with the ICAT architecture to allow passing user actions between the simulation and ICAT?
- Can the product be enhanced to allow the ICAT to stop bad user actions?
- Can new custom control be added to the product?
- Can the personal trainer interface be created using the product?
- Can the product create multi-window simulations?

- Can the product support SHIFT-like "navigation" windows that call up other panels on demand?
- Can the product support CRT displays in panels?

Two potential cross platform simulation development systems have been identified that may enable the researchers to build SWEs similar to the one used in SHIFT: Lab View 2 and Virtual Application Prototyping System (VAPS).

Task 2: *Extend OPL To Support Common Mistakes and Their Recovery Procedures*. During the SHIFT project, OPL greatly simplified the creation of exercises. However, in developing SHIFT, the researchers identified a limitation in OPL. In some situations, an instructional designer may want the ICAT system to ignore a student's mistake. This "leash" allows the student to discover and correct the error himself. The second task of this project is to extend OPL to support leashes. Extensions to the OPL language itself will be defined and tested against previous ICAT requirements.

Although OPL provides an efficient way of specifying exercises, each exercise must be translated by hand into the data files required for implementation. The goal of this task is to provide a complete set of requirements for an "Exercise Editor" to be built in Phase II. The exercise editor will facilitate both drawing OPL diagrams and generating the data files.

Task 3: *Develop an Integrated Technique for Editing Misconception Information*. The new technique must allow specifying (1) concepts, misconceptions, and messages; (2) associations between concepts, misconceptions, and messages; and (3) behavior patterns which imply each misconception and message.

This remediation mechanism was used with great success in the IPT. However, like OPL notation, the misconceptions must be translated by hand into the tree notation used. A primary goal of this task is to provide the requirements necessary to build a "Misconception and Message Editor" during Phase II.

Task 4: *Extend the Exercise Selection Algorithm To Support Structured Lessons*. The exercise selection algorithm currently supports choosing a base lesson. Once inside the lesson, though, it has no provisions for selecting different exercise scenarios. The researchers intend to develop a notation similar to OPL for use in describing lesson sequences. Once the notation is complete, it will be integrated into the exercise selection algorithm.

Like OPL and misconceptions, these meta-data indices must be translated by hand into executable data files. The goal of this task si to provide a complete set of requirements for a "Lesson Plan Editor" to be built in Phase II.

Potential Follow-On Work

In Phase II researchers will develop a complete set of modules for authoring ICAT systems. These modules will include the exercise editor, the misconception and message editor, lesson plan editor, student history, teacher's window, data dictionary, and ICAT supervisor.

14.4.9 ICAT Authoring Environment

After reviewing previous ICAT implementations and relevant literature, the STB has selected synergistic implementations for each ICAT module. A layer of graphical modeling notation has been layered on top of each module. These notations enable an instructional designer to easily diagram a new ICAT design.

While these notations greatly aid in specifying ICAT knowledge, someone with programing knowledge of these particular algorithms must still hand translate the diagrams into computer data files. To achieve the project goal of constructing an authoring environment that helps the instruction designer to create a new ICAT system, this environment will provide graphical editors for each modeling notation, and automatically translate diagrams into the required data files. In addition, these editors will be integrated with the run-time algorithms. This will allow the editors to also act as debugging tools. While student is working an exercise, the editors will automatically highlight the ICAT diagrams to show the student's progress.

The chosen approach to building an ICAT system requires simulating the work environment and modeling a personal trainer. In previous NASA ICAT systems, the SWE has been built *ad hoc*, using whatever methods where known to the implementor, and little has been done to clarify or simplify SWE construction. Accordingly, the STB minimized the interaction between the SWE and the other modules of the ICAT system.

Development Status Authoring environment under development.

Architecture The authoring environment plans elaborate on NASA's second generation ICAT architecture; see Figure 44 on page 199. NASA's architecture, shown in the diagram as the lower level, defines which modules are necessary to build an ICAT. These modules include the Action Evaluator which objectively evaluates each operation the student performs; the Remediation which subjectively decides what to tell each student based on both the student history and the "teaching model"; the Lesson Planner which chooses the most appropriate exercises for each student; and the simulation which models the environment for which the student is to be trained. This architecture does not specify how to implement each module. Different methods of implementation can be used.

In addition to simulating the work environment, building a system on the architecture requires modeling a PT. Rectangular modules represent sections of the PT. A PT has three main duties, assigned as discussed below.

Action Evaluator Once an SWE is defined, the instructional designer must develop exercises to be performed in the environment. The two types of exercises are (1) those that present new material, and (2) those that apply the material. These concept application les-

sons also serve as the way the ICAT system evaluates the student's progress. The OPL graphical notation aids the designer in specifying what things a student should do during a lesson. The Action Evaluator compares student behavior to this specified behavior. The SWE only interacts with the PT through the Action Evaluator and the Student History.

Remediation

The major part of trainer modeling is watching the series of student actions and identifying patterns that signify a misunderstanding of the material. It is based on the "Misconception Theory" that holds that teachers can easily describe what common mistakes students make, what misconceptions cause these mistakes, and how to remediate them. An efficient matching algorithm stores patterns of behavior and maps them to misconceptions. The algorithm provides for structured relationships between misconceptions that allow the system to fallback and remediate a more general concept if several related misconceptions have been diagnosed. It stores one or more types of remediation for each misconception, allowing explanations to be chosen based on the student's background.

Lesson Planner

The misconception theory allows the PT to select appropriate exercises for each student. Exercise selection is based on knowledge the PT has gained from watching the student complete previous exercises. Selection requires indexing meta-knowledge (concepts that must be understood to successfully complete the exercise and common misconceptions) about each exercise. This meta-knowledge is used to compile a list of applied concepts and diagnosed misconceptions, and these are used to develop a list of recommended concepts for future study.

Other Modules

The other three modules of the system provide basic support to these core routines. The Student History remembers what the student did, what the student knows, and what he was told. The Supervisor provides the PT's graphical interface with the student and coordinates execution of the ICAT. The Decoder acts as a mini database for the ICAT. It maintains a list of all actions, concepts, exercise names, messages, and misconceptions used in a particular ICAT.

Evaluation Status

Each module of the ICAT authoring environment will be delivered to the Space Flight Training Division as it is completed. There are no formalized evaluation plans at this time. However, feedback from training personnel will be incorporated into future revisions of each tool.

Operating Environment

The ICAT authoring system is being developed to run on Macintosh, under windows on PCs, and under X-windows on Unix platforms.

Future System Development Plans

Based on use of the architecture by the Space Flight Training Division, an ongoing program of refinement and maintenance will be put in place to ensure the long-term usability and success of the product.

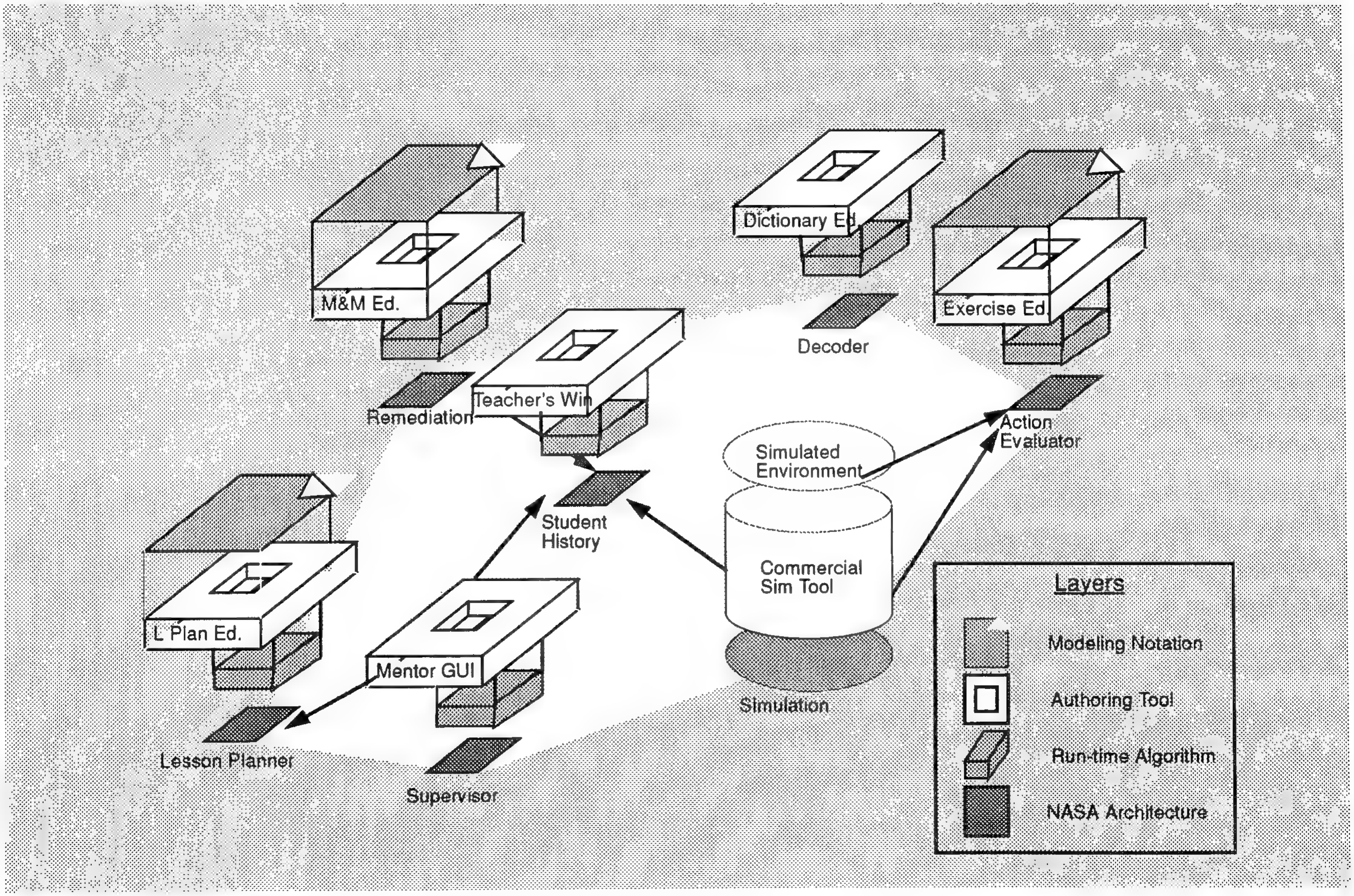


Figure 44. Overview of NASA Authoring Environment Architecture

14.4.10 Advanced Training Technologies Project

The objective of this project is to enhance the effectiveness of NASA training through the development and application of advanced technology. Systems will be implemented to achieve the following goals:

- Incorporate student evaluation techniques into instructor unique tools.
- Continue development of a practical ICAT shell for instructorless training.
- Develop field deployable training tools for "just-in-time" training.
- Enable the use of virtual environments as an interface to training systems.

This project offers some specific cost benefits. Training needs will be decreased by up to five Equivalent Person/year, and ground training costs decreased by $900K per year. Additionally, the VR system could payback $2M towards a Space Station & Shuttle Mission Simulator visual upgrade.

Programmatic Background

This project began in 1987 and will complete in 1996. The funding amount for FY94 is approximately $3M.

STB researchers will be supported by active participation from DT/Space Flight Training Division at JSC. Active coordination with user communities in Mission Operations Directorate, Flight Crew Operations Directorate, and other NASA field centers will be maintained as appropriate.

Planned Products

The following products are expected from this effort:

- ICAT applications for Mission Operations Directorate.
- Support for technology transfer and ICAT system maintenance for Mission Operations Directorate.
- Development (with DT/Space Flight Training) of software tools that enable trainers to build and maintain ICAT systems.
- Virtual environment training applications for Mission Operations Directorate.
- Integration of ICAT systems with virtual environments.
- Shared virtual environments for collaborative training with other NASA centers and international partners.
- Software tools that enable trainers to build and maintain complex virtual environments for training.

Approach

This effort involves the following tasks:

- Continue refinement of the ICAT architecture and TARGET.
- Place an ICAT system on-board a space shuttle flight through a Detailed Test Objective (DTO) to begin field deployable training evaluation.
- Integrate the ICAT architecture with a virtual environment and develop a comprehensive application.

- Develop a comprehensive software environment for building and maintaining complex virtual environments for training.
- Develop distributed ICAT systems to support concurrent training of multiple personnel.

The major milestones are depicted in Table 6 on page 201.

Table 6. Advanced Training Technologies Milestones

Milestones	FY94	FY95	FY96	FY97	FY98	FY99
ICAT v2 Tools		X	X			
ICAT Applications			X	X	X	
ICAT/VE Integration		X				
ICAT/VE Applications			X	X	X	
On-Board ICAT DTO		X	X			
On-Board ICAT/VE DTO			X	X	X	
Training Support		X	X	X	X	X

Accomplishments

FY94 accomplishments include the following

- Hubble Space Telescope Virtual Environment used to train STS-61 Flight Control Team.
- Integration of ICAT capabilities with the Hubble Space Telescope virtual environment.
- Version 2.0 of ICAT architecture design completed.
- Prototype of Version 2.0 of TARGET completed.

15. NATIONAL SCIENCE FOUNDATION

15.1 Mission and Role of the EHR, RED

The National Science Foundation (NSF) Directorate for Education and Human Resources (EHR) aims to improve the quality of science, mathematics, engineering, and technology education for all students at all levels. It works to increase the number of individuals—from all backgrounds—who pursue advanced studies and careers in science and technology. EHR supports scientific literacy in schools and colleges and also supports informal science education programs that create public awareness of, interest in, and understanding of scientific and technological developments.

The Division of Research, Evaluation, and Dissemination (RED), part of EHR, was formed in 1992 to better integrate and provide a focus for these activities, each of which is essential to EHR as a whole. RED not only contributes to the broad field of educational research and improvement by funding projects through grants, contracts, and cooperative agreements but also provides conceptual and technical assistance to the various EHR programs and the principal investigators they support.

15.2 Background to the Advanced Technologies Program

The work discussed here is sponsored by the Applications of Advanced Technology (AAT) program, one of RED's basic and applied research programs. AAT is an R&D program concerned with issues at the forefront of technology. It seeks proposals that focus on new, high-risk, high-gain applications of technologies to advance the learning and teaching of science, mathematics, and engineering at all levels of education. It has two complementary goals [NSF 19930]:

- To lay the research and conceptual foundations that will advance the field, explore new paradigms, extend and establish new human limits, articulate new technology-based science and mathematics, and determine the strengths and weaknesses of new, innovative applications of advanced technologies to education.

- To lay the foundations and knowledge necessary for planners, policy makers, and educational decision makers to design new systems and establish the conditions necessary to introduce revolutionary computer and telecommunications systems and related technologies—the goal being to bring science and mathematics into classrooms, schools, homes, and other significant places of learning at an accelerated pace.

Six areas of particular concern or opportunity have been identified. Proposals addressing these concerns and opportunities are particularly encouraged. These areas are as follows:

- Knowledge-based systems and intelligent tutors.
- Intelligent tools.
- Production and authoring systems.
- Problem-solving and programming.
- Telecommunications, networks, and educational infrastructures.
- Educational network testbeds.

Preliminary proposals may be submitted for this program at any time. After completing the preliminary proposal stage, formal proposals may be submitted at one of two annual target dates. Funding decisions are typically made about six months after the formal submission.

15.3 Summary of Past ITS Work

Information on past ITS-related efforts sponsored by NSF is not available.

15.4 On-Going ITS-Related Tasks

NSF is currently building an index of their on-going projects. In the interim, the projects discussed below were identified from the 1992 abstracts of funded projects [NSF 1992].

15.4.1 Methodology for Developing a Practical Algebra Tutor Project

The Algebra 1 curriculum is one of the most problematic courses in the city of Pittsburgh. Students have been working with the Forester text, and 70% of the students have received grades of D's and E's. This is with about one-third of the students streamed into a non-academic mathematics track and so not in Algebra 1. A decision was made to

greatly de-emphasize the symbol manipulation content of the course and concentrate on a functional approach which teaches students to use linear and quadratic equations and inequalities to model real-world situations. Much of the traditional content of Algebra 1 is postponed to Algebra 2 where it is retaught anyway. The goal is to make sure that students really master this basic material and are able to apply it in real-world situations. These decisions were made as part of the Pittsburgh Urban Mathematics Project (PUMP).

The goal in the effort discussed here is to define what is involved in taking the cognitive tutor methodology and using it to implement a course. Researchers have taken as their task to develop a tutor that will support the city of Pittsburgh PUMP Algebra curriculum. This is an attempt to teach algebra to all ninth grade high school students in a way that is in keeping with National Council of Teachers of Mathematics (NCTM) standards and which is meaningful to students raised in an urban setting. It focuses on teaching students mathematical tools to solve realistic problems. The tutor guides students in standard symbol manipulation, use of graphing facilities, and spreadsheet facilities. The tutor will be used by students two days a week for a total of about 70 classroom sessions and students will go through progressively more difficult problems at their own pace. The city has currently created classrooms of 25 Quadra 610's in 3 of its 12 high schools to deploy this curriculum and has plans for adding additional high schools.

Programmatic Background This effort is being undertaken by Carnegie Mellon University, Department of Psychology, and the Principal Investigator is Prof. John Anderson. The project began in August 1992 and is due to complete in January 1995. The funding is provided under NSF Grant MDR 92-53161, an amount of approximately $790K. A second two-year effort is awaiting approval.

Planned Products Prototype Algebra 1 Tutor.

Prior Work[1] The researchers developed production system models in ACT of how students solved problems in LISP, geometry, and algebra. Computer tutors were developed around these cognitive models. Construction of these tutors was guided by a set of eight principles loosely based on the ACT theory. Early evaluations of these tutors usually, but not always, showed significant achievement gains. Best case evaluations showed that students could achieve at least the same level of proficiency as conventional instruction in one-third of the time. Empirical studies showed that students were learning skills in production-rule units and that the best tutorial interaction style was one in which the tutor provides immediate feedback, consisting of short and directed error messages. The tutors appear to work better if they present themselves to students as non-human tools to assist learning rather than as emulations of human tutors. Students working with these tutors display the capa-

[1] This material is taken from [Anderson 94a].

bility to transfer to other environments to the degree that they can map the tutor environment into the test environment.

These experiences have coalesced into a new system for developing and deploying tutors. This system involves first selecting a problem-solving interface, then constructing a curriculum under the guidance of a domain expert, designing a cognitive model for solving problems in that environment, building instruction around the productions in that model, and finally deploying the tutor in the classroom.

Approach

In the first year, researchers developed a tutor for integrating use of a spreadsheet, graphic routine, and symbol-manipulator to solve problems. This was directed at the core of the curriculum. The researchers implemented about a month's worth of material, delivered in three- to five-day blocks throughout the year. Over the period of their work with the tutor, students got to the point where they were able to solve problems involving systems of linear equations. This work was consistent with what they were doing otherwise in their classroom. It provided essentially an opportunity for students to get extensive practice on the basic concepts. In the first year, the tutor was always used after the material had been introduced in class.

Now in the second year, researchers are extending this facility to quadratic equations and linear inequalities and are developing a facility for teaching the symbol manipulation involved in solving of equations—a topic which the teachers feel cannot be abandoned. Over the summer and in the beginning of this academic year, the tutor has been used with some success as a first introduction to the material. The researchers are working towards a situation where students will spend approximately half of their time on the tutor, with the other half of the time in the classroom, much of that involved in group projects. They also are working towards having the tutor interface be a tool for these group projects.

When students are working on the tutors, the teachers circulate around the classroom providing students with individual instruction. The tutor does not require students to solve problems in specific ways, and one thing teachers encourage is student's solving problems in their own ways. The researchers are working towards creating an easy problem entry mode in which students can enter their own problems and be tutored on their solution.

Potential Follow-On

The major focus of follow-on work will be in evaluations that provide more controlled studies of achievement and better understanding of the motivational differences that are reported in the tutored classrooms.

The currently feasible evaluation is one where the curriculum and teachers are held constant and only the presence of the tutor is manipulated. The first opportunity to do this will occur in the second semester of 1994. The researchers will have in place about two months of tutor material for which the students will be ready in the classes using the Hadley material. In the first two months (February and March) of the semester, half of the classes will get exposure to this material while the other classes progress on non-tutor material. Then this will switch for the next two months (April and May). This situation is necessitated by the fact that all the classes have to go through the same laboratory. By comparing achievement performance of the classes at the end of March and at the end of May, researchers will

be able to assess the value added of the tutors to the Hadley curriculum. In the future, as adoption of the Hadley curriculum outstrips the availability of computers, they will be able to repeat this test in other schools.

Examples of some of the issues to be investigated answered are as follows:

- Identification of the pertinent motivational factors and how can they be facilitated.
- Teachers' attitudes about the tutors.

15.4.2 Algebra Tutor

Figure 45 on page 208 provide a sample screen taken from the Algebra 1 tutor.

Development Status Prototype tutor under development.

Evaluation Status All of the evaluations of the algebra tutor completed to date are preliminary in character. However, the results are positive. The results are preliminary for four reasons. First, the researchers are just now getting the schools to buy into the project to the point where they will allow well-designed evaluations to be performed. Second, the researchers are still amassing enough tutor software to be able to see how large an effect the tutor can have. Third, they still are identifying the issues that need evaluation. Fourth, no one is funded for doing third-party evaluations. The following repeats evaluation results reported in [Anderson 94b].

Grades: At Langley Elementary School, Pittsburgh, through the third grading period, 55-60% of students in the pilot classes were receiving grades of A, B, or C. Only 40-45% of students in the non-pilot classes were receiving similar grades. Traditionally across the district, 30-40% students in Algebra 1 receive these grades.

Iowa Algebra Aptitude Test: This test, designed to measure a student's algebra aptitude, was administered to all the algebra students at Langley, all the geometry students at Langley, and to both algebra and geometry students in several other Pittsburgh High Schools at the beginning of the school year. The mean score for all Langley algebra students was approximately 24, and for all geometry students it was approximately 30. When the test was administered at the end of the year, the non-pilot algebra students scores rose to 30.5 and the pilot students scores were 33.5, which were a significant difference.

SAT Test: The project teacher also administered a set of questions from the mathematics section of the new SAT test to his students at the end of the year. Using these results to predict student SAT scores, the one class average score was 380 and the other class average score was 410. The city-wide SAT average math score for juniors and seniors has, for the last several years, been in the 375-400 range. So the pilot students, after completing only one year of high school mathematics, performed at about the average for all high school students.

Teacher Attitudes: The pilot teachers were so enthusiastic about this first experience, they convinced their colleagues at Langley to eliminate Problem-Solving

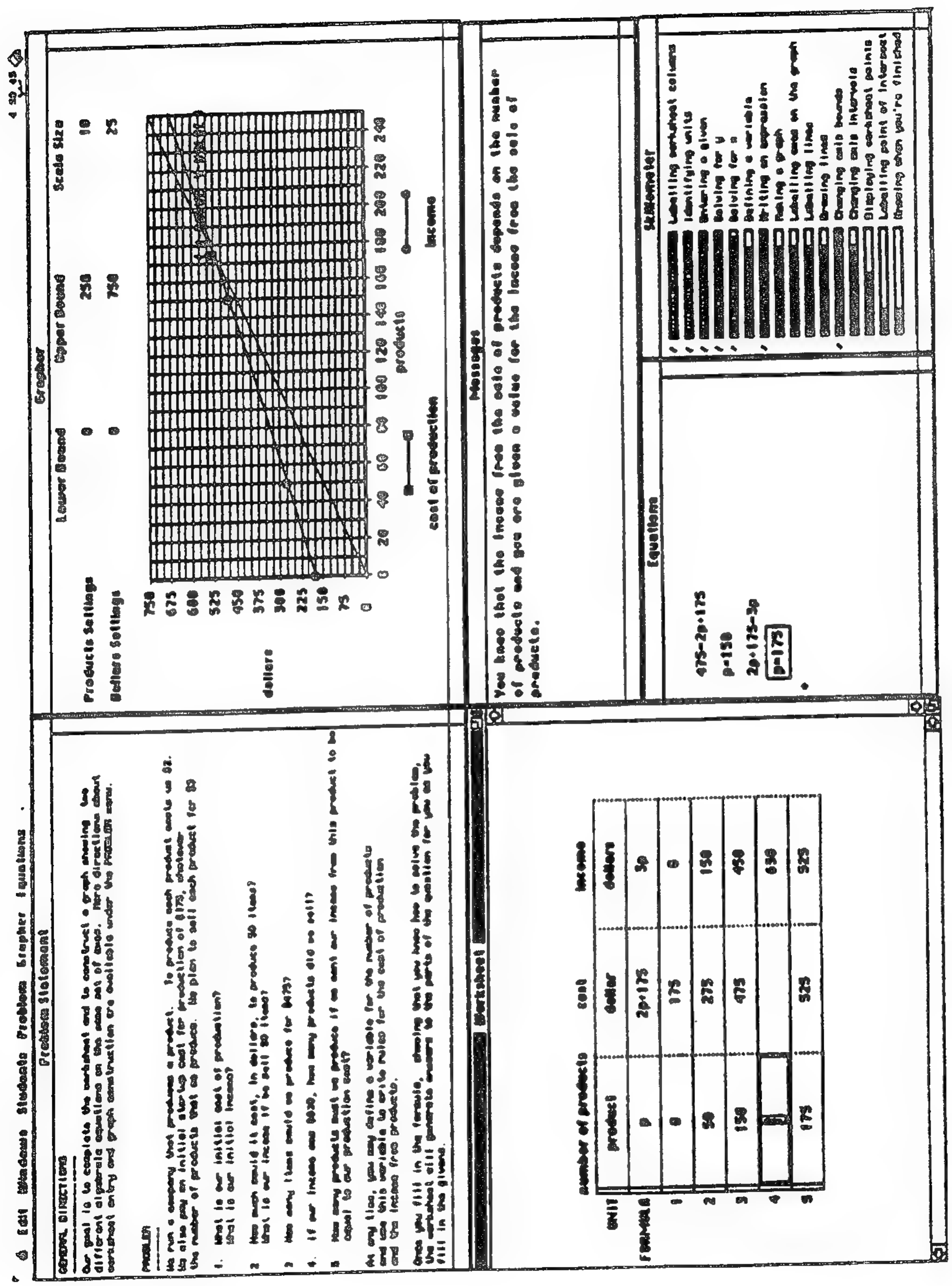


Figure 45. Sample Screen from the Algebra 1 Tutor

One from the course offerings and to require all students entering Langley to enroll in the PUMP Algebra course next year.

Student Attitudes: An attitudinal survey given to the students at the end of the school year indicated that the majority of the pilot students had a positive experience in Algebra and most cited the tutors and the curriculum for the reason. Additionally, many commented on the structure of the classroom and the fact that this was the "one class where we had to think." Students would come to class early and leave late on those days when they were using the tutors, and many others would stay after school, often until 4:00, working on the tutors. Another reflection of student attitudes was that many students who were not engaged in the regular classroom were "glued to their monitors" in the computer lab.

Final Exam: In light of the recently released findings about students' inability to do extended open-ended questions, the PUMP students were at least twice as successful in adequately answering this type of problem.

System Development Plans

There are two overlapping directions for future system development.

Completing Tutor Suppor

Tutor support for the existing Algebra 1 curriculum will be completed. This means complete support for problem-solving involving linear and quadratic equalities and inequalities. It also means complete support for equation solving involving systems of linear and quadratic equations. This would give the tutor a curriculum to cover what would amount to about 50% of the Algebra 1 course and enable the researchers to assess what the consequence of this extensive tutoring experience would be for mathematics achievement. This evaluation of the tutoring methodology is absolutely essential.

Additional Facilities

Additional facilities planned for the tutor include (1) a teacher's interface that allows teachers to enter problems, select the curriculum, determine the mode of interaction, and modify student models; and (2) a dialog system that allows students to construct "free form" answers to questions. While engaging in unrestricted natural language would be a bottomless pit, the structured editor technology that the researchers have developed with their programming tutors can be used to let students construct responses to questions. This is important because a clear deficit in the current tutors is that students do not convert their solutions into the kinds of descriptions one sees in written reports.

15.4.3 Using Automatic Speech Recognition To Improve Reading Comprehension Project

Much software exists to teach reading but it is limited in its ability to listen and/or intervene. This project is intended to address such limitations by adapting continuous speech recognition to listen to children read connected text, automatically triggering pedagogically appropriate interventions. More specifically, the goal of this two-year pilot study is to test the technical feasibility of using automatic speech recognition to improve reading

comprehension. The key research issues to be studied are (1) what should a reading coach do, and (2) how should speech technology be adapted to it. In this way, the study will discover how modern technology can be brought to bear in the struggle against illiteracy.

Researchers are focusing on children in grades one through three because that is where oral reading is emphasized, and are focusing on the children that most need help. This work also has the potential to benefit a number of other populations. It should help older children and adults who have not learned to read. It should help children and adults learn English as a second language. It should enable children who would learn to read satisfactorily to read at a younger age, or read and comprehend much more advanced material than they would otherwise. Similarly, the methods developed in this work should help people learn foreign languages. Results to date support the plausibility of the approach, both technically and pedagogically.

At the current time, the reading coach is a very basic intelligent tutor. The student model is a simplistic one of words missed. Although the response is individualized to what the reader did, the response is not a rich one. In future work, researchers hope to extend these aspects of the coach. Their intention is build up the student model as more understanding is gained about patterns of mistakes. A scaled-up version of the coach is needed to enable the necessary data collection.

Programmatic Background This work is funded under NSF Grant MDR-9154059. The grant began in May 1992 and is scheduled to complete in mid 1994. The funding amount is for approximately $870K. The work is being performed by Carnegie Mellon University, Robotics Institute. The Principal Investigator is Dr. Jack Mostow.

Planned products Prototype Emily coach.

Prior Work Initial work was funded by MCC in early 1992 and involved language modeling using Sphinx-I.

Additionally, the current effort exploits the research that built the CMU Sphinx-II speech recognizer. However, analysis of oral reading differs from speech recognition in an important way. In speech recognition, the problem is to reconstruct from the speech signal what sequence of words the speaker said. In contrast, the problem addressed in this research is to figure out, given the text, where the speaker departed from it.

Approach The approach is as follows.

Task 1: *Formulate Suitable Interventions.* **This task has been completed**. To evaluate the pedagogical efficacy of possible interventions independent of the accuracy of the speech recognizer, researchers developed a "Wizard of Oz" (WOZ) simulation of the reading coach. The WOZ simulated coach appeared real to the subjects, but was controlled behind the scenes by a human experimenter. The experimenter's

role consisted of listening to the reader, following along in the text, and marking each word as correct or misread. The rest of the WOZ was automatic, and used the marking information to trigger its interventions.

The initial design of the WOZ was based on observation and analysis of individual assistance provided by human reading experts (principally Dr. Leslie Thyberg). The most frequent forms of intervention that appeared feasible to automate were identified and automated in an incremental fashion intended to facilitate rapid design iteration. At first the interventions were simply written instructions for the human experimenter to follow when helping children to read. As they stabilized, the interventions were implemented in code allowing experimenter selection of an intervention via a menu. Based on further experience, the triggers for these interventions were automated. For example, the trigger for the SaySentence intervention fires when the reader reaches the end of a sentence after misreading three or more words in it. As the triggers were automated, responsibility for selecting interventions shifted from the experimenter to the WOZ. Automating the triggers made explicit the conditions for invoking the interventions and maintaining a natural mixed-initiative flow of interaction between reader and coach. At this point, the experimenter was out of the loop except for marking each word as correct or misread.

Task 2: *Using Sphinx-II Speech Recognizer To Follow Reader and Detect Missing Words.* **This task has been completed**. To automate the speech recognition capabilities needed, researchers adapted Carnegie Mellon's Sphinx-II speech recognition system to do two tasks for which it wasn't designed. One task was to try to follow the reader. The other task was to identify which words the reader missed.

The resulting system, implemented in Fall 1992, was named Evelyn [Mostow 93]. It displayed a page of text on the screen. As the user read aloud, Evelyn followed along, highlighting the current word, though it lagged behind real time. At the end of the page, Evelyn simply highlighted and spoke the words it thought the reader missed. This sort of feedback is not appropriate for young readers; it was intended simply to demonstrate Evelyn's speech capabilities. Evelyn's accuracy in detecting misread words was evaluated on a corpus of children's oral reading.

Task 3: *Initial Integration of the Automated Speech Analysis with Simulated Coach.* **This task has been completed**. The next task was to automate the coach by using automated speech analysis to replace the human operator in following the reader and flagging misread words. The resulting system, named Emily, was implemented in Spring 1993 and first demonstrated at the June 1993 Project Directors' meeting of the NSF Applications of Advanced Technology program. Unlike Evelyn, which waited until the end of the entire page to respond, Emily responds after each sentence.

Task 4: *Adjust Automated Coach To Both Reduce and Compensate for Recognizer Errors.* **This task is ongoing**. Since the interventions used by the coach were tested and refined in the context of the simulated coach, triggering them based on imperfect automatic speech recognition requires addressing the effect of speech recognition errors. One problem is to reduce these errors by providing better inputs to the

speech recognizer. Since perfect accuracy is impossible, another problem is to modify the coach's interventions to behave gracefully even when the recognizer makes mistakes. For example, when the coach asks the student to reread a word, it can never say the student was right or wrong, since it can't be certain. Instead, the researchers modified the coach to echo the correct word after the student rereads it. This feedback is intended to be either confirmatory or corrective, depending on whether the student was correct.

Researchers plan to test the overall pedagogical effectiveness of the automated coach shortly. One goal is to test whether (like the simulated coach) it enables children to read and comprehend more advanced material than they can read on their own. Another goal is to identify its limitations and their causes. In particular, how do Emily's current level of accuracy and speed affect its pedagogical effectiveness?

Potential Follow-On

The researchers propose a new way to combat illiteracy by harnessing advanced technologies: getting computers to listen to children read aloud. The pilot study has already found initial evidence for the technical feasibility and pedagogical effectiveness of this approach. Early evidence suggests that it may enable children to comprehend material as much as two years above their independent reading level.

The researchers propose to build on this success by exploring a number of new directions:

- To help children read, they plan to analyze oral reading faster, more accurately, and more informatively, and to use this analysis to provide better assistance.
- To help children learn, they plan to provide weeks or months of computer-assisted reading about science, and to analyze how such assistance affects learning.
- To fit individual needs, the researchers plan to vary text dynamically, so as to clarify confusing points, adjust the level of difficulty of the text, and provide individually tailored examples of what the child needs to learn.
- To help children write, they plan to let them dictate stories and edit inaccurate transcripts produced by automatic speech recognition.

The scientific effect of success would be significant: speech communication would help students and automatic tutors to understand each other much better. The social effect would be enormous: cutting illiteracy by even 10% would save the nation over $20B a year.

15.4.4 Emily Coach

Emily is designed to help a child read and comprehend a given story. It is intended to maintain a fluent, pleasant reading experience that gives the child practice in reading connected text, plus enough assistance to be able to comprehend it. To achieve this goal, Emily uses a combination of reading and listening that researchers have named "shared

reading," where the child reads wherever possible, and the coach helps wherever necessary [Mostow 94].

Emily intervenes when the reader misreads (fails to speak the correct word) one or more words in the current sentence, gets stuck, or clicks on a word to get help. The system does not treat hesitations, sounding out, false starts, self-corrections, or other insertions as misreading. Emily's current set of interventions target two obstacles that interfere with children's reading comprehension. These obstacles are word identification and the problem that, having spent so much of their attention figuring out a word, readers may not comprehend the overall meaning.

It should be noted, however, that the ability of this system to *listen* to a reader is its most novel aspect. Although as yet unmeasured, the effect that listening has on motivation and cognition is believed to be substantial. Figure 46 on page 213 shows a picture taken of Emily in actual classroom use.

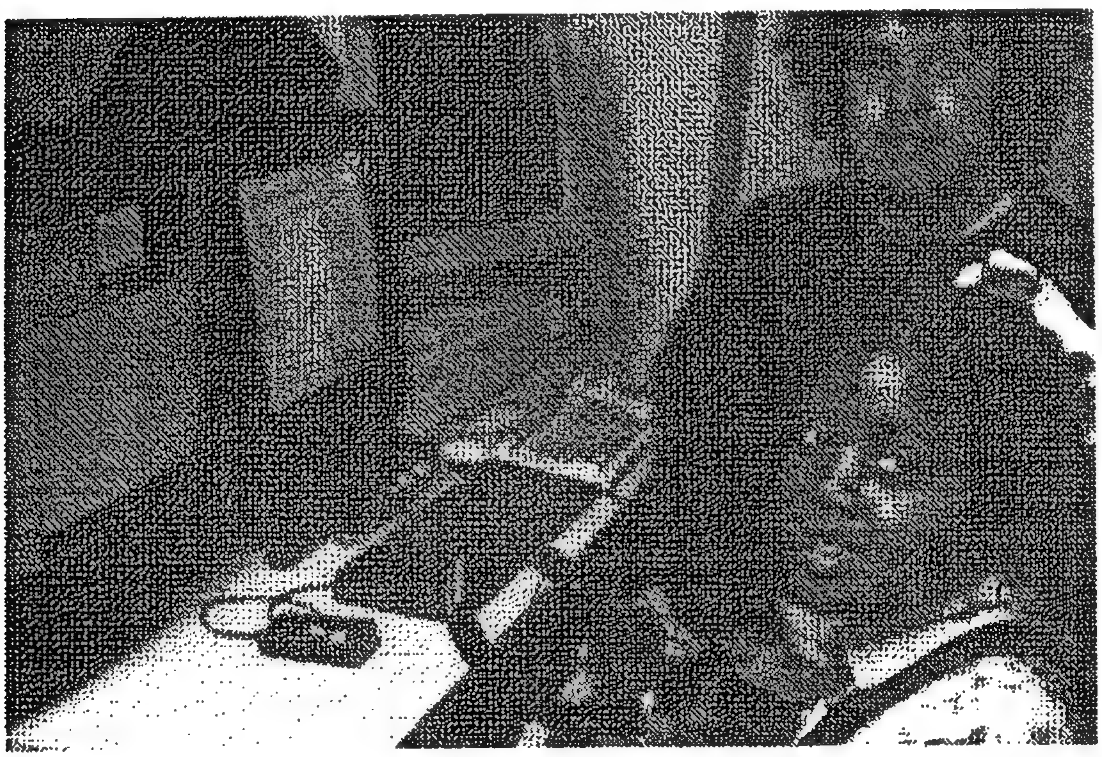


Figure 46. Emily in Classroom Use

Development Status Emily is a prototype constructed as a research tool.

Architecture

Emily consists of two components, the Intervenor and the Speech Recognizer, as described below.

Intervenor

The Intervenor interacts with the reader and the Speech Recognizer. It tells the recognizer where in the text to start listening—either at the beginning of a new sentence, after a word is spoken by the coach, or at a word the coach has just prompted the reader to reread. Four times a second, the recognizer reports the sequence of words it thinks it has heard so far. It detects which words of text were misread by aligning the output of the recognizer against the text. It detects when a reader reaches the end of a given fragment of text by checking if the recognizer has output the last word of the fragment. The Intervener assumes that the reader is stuck on a word if the time limit for progressing between words is exceeded without a previously unread word appearing in the recognizer output. The Intervener is invoked whenever the reader reaches the end of a sentence, gets stuck, or clicks the mouse on a word for help.

Speech Recognizer

Emily is built on top of the Sphinx-II speech recognizer. Sphinx's input consists of digitized speech in the form of 16,000 16-bit samples per second from a microphone via an analog-to-digital converter. Its output consists of a segmentation of the input signal into a string of words, noises, and silences. Sphinx uses three primary knowledge sources: a database of phonetic Hidden Markov Models, a dictionary of pronunciations, and a language model of word pair transition probabilities—as well as several parameters that control its Viterbi beam search for the likeliest transcription of the input speech signal. The recognizer evaluates competing sequences of lexical symbols based on the degree of acoustic match specified by its phonetic models, the pronunciations specified by its lexicon, and *a priori* probability specified by its language model. Thus Sphinx-II's recognition accuracy is limited by how well these representations model the speech input. Emily uses 7,000 phonetic Hidden Markov Models trained on 7,200 sentences read by 84 adult speakers (42 male and 42 female).

Evaluation

To date, only preliminary pedagogical and technical testing has been conducted.

Pedagogical Evaluation

The primary purpose of the initial evaluation was to test the overall effectiveness of the interventions. It was conducted in November 1993. The hypothesis being tested was that the interventions would enable struggling readers to read and comprehend material significantly more advanced than what they could read on their own. The subjects were 12 public school second graders that had been identified by their teachers as having difficulty reading. Evaluation materials were taken from Spache's *Diagnostic Reading Scales*. Subjects were compared for independent reading level, coach-assisted reading level, and listening comprehension level. The test used a simulated coach.

The hypothesis was supported, with the subject's assisted reading level higher than their independent reading level by an average of 0.6 years. The interventions also reduced frustration, with the children misreading only 2.6% of the words with assistance, as opposed to 12.3% of the words without assistance.

Technical Evaluation

Technical evaluation is being performed repeatedly. The latest evaluation was conducted in April 1994. The purpose of this evaluation was to assess accuracy of the

speech recognizer in detecting mistakes. Fifteen second graders were selected for subjects and these children were tested on 514 sentences. The average disfluency was 1.6% words missed (+6% given words). The system detected 49% of missed words and raised false alarms for less than 4% of correct words.

Operating Environment

Emily's speech processing consists of some signal processing performed on a NeXT computer in close to real time, along with the coaching system. The speech recognition is performed on a DEC 3000 or HP 735. For natural speech quality, Emily normally outputs predigitized human speech. However, two synthesized voices (ORATOR and DecTalk) are available as alternatives. Within three years, researchers expect the full system to be available on a $3K workstation.

Future System Development Plans

The major focus of the continuing development is to improve accuracy and speed, and scale Emily up to work on larger pieces of text. Researchers plan to improve Emily's flexibility by reimplementing its language model to take constant space instead of space proportional to (or even quadratic in) the total amount of text. Eliminating the need to reload language models for different text could enable Emily to monitor oral reading of text generated on the fly. Pending collection of a large corpus of children's oral reading, researchers plan to adapt adult phonetic models to work better on children's speech by using an interpolative training method.

15.4.5 Developing Flexible Explanation Generators for Intelligent Tutoring Systems Project

A major limitation of current advisory systems (for example, expert systems and ITS) is their restricted ability to explain. The goal of this research is to develop and evaluate a *flexible* explanation facility, one that can dynamically generate responses to questions not anticipated by the system's designers and that can tailor these responses to individual users. To achieve this flexibility, the researchers are developing a large knowledge base, a viewpoint construction facility, and a modeling facility.

In the long term, the researchers plan to build and evaluate advisory systems with flexible explanation facilities for scientists in numerous domains. In the short term, they are focusing on a single complex domain—biological ecology—and will build and evaluate an advisory system with a flexible explanation facility for freshman-level students studying this particular area of ecology.

Programmatic Background

This project is being conducted by researchers at the University of Texas, Department of Computer Science, under NSF grant number IRI-9120310. The project started in April 1992 and is due to complete in March 1995. The total grant amount is $380K. The Principle Investigator is Dr. Bruce Porter.

Planned Products

The primary results of this research will be (1) a tutoring system for a core topic in college-level biology; (2) a critical evaluation of the system based upon its use

in an introductory biology course at the University of Texas at Austin; and (3) general methods and tools for developing similar tutoring systems in other domains.

Prior Work

During the last six years, the researchers have encoded a significant amount of scientific knowledge in one area of college-level biology to produce a very large knowledge base. (Containing over 100,000 facts, the knowledge base is one of the largest of its kind.) They have also developed prototype systems for coherently explaining this knowledge to students.

Approach

Researchers are developing a three-part solution to the problem of automatically generating explanations in an ITS:

- Selecting and organizing knowledge with viewpoints and models. The first part of the solution is methods for structuring collections of facts in a knowledge base. One organizing structure is that of *viewpoints*, which provide coherent descriptions of objects or processes. For example, the viewpoint "photosynthesis as a production process" selects and organizes facts to explain how photosynthesis produces glucose from carbon dioxide and water. Another organizing structure is that of *models*, which are built from viewpoints and support computer simulation and visualization. For example, an energy flow model of the plant includes the viewpoints "photosynthesis as an energy transduction process" and "respiration as an energy transfer process," and it allows a tutoring system to explain and predict the effects of changes in light wavelength on a plant's photosynthetic or respiratory rate under a variety of specific circumstances.
- Automatically generating new viewpoints and models on demand. The second part of the solution is methods to automatically generate *new* viewpoints and models. This ability is important because system designers cannot anticipate all the viewpoints and models necessary for effective tutoring. The tutoring system will be able to construct these viewpoints by selecting and reorganizing the individual facts representing existing viewpoints in the knowledge base.
- Generating explanations which relate new information to what the student already knows. The final part of the solution is a set of methods to automatically generate *integrative explanations*, which explicitly relate new information to what the student already knows. This is important to tutoring systems because the coherence of an explanation depends upon the teaching situation. The tutoring system will record the discourse with each student and will explain new topics in ways that relate to that student's knowledge and interests.

Status

During the course of this research project, the researchers have accomplished the following. First, they have significantly extended their knowledge base on the anatomy and physiology of plants. This fine-grained, semantic net knowledge base is one of the largest of its kind, currently containing about 175,000 facts. It provides a laboratory for research on many tasks in AI, such as knowledge representation, question answering, and, of course, intelligent tutoring.

Second, they have built a prototype system for explanation generation, the task of extracting information from a knowledge base to answer a question with a coherent explanation. Unlike most other explanation generators, this tool is quite robust; it is capable of generating countless explanations, adapted to a model of the user, from all parts of the knowledge base. After selecting the content of the explanation, the system uses the FUF text generator (from Columbia University) to produce English text.

Third, the researchers have built a prototype system for answering prediction questions ("what-if" questions) concerning physical devices. The basic AI task performed by the system is compositional modeling, the task of building a device model that can be simulated to predict the device's behavior in a particular scenario. The models constructed by the system are simulated by QSIM (from related work at the University of Texas at Austin).

Potential Follow-On This work has drawn on the attention, and funding, of Digital Equipment Corporation (DEC), Marlboro, MA. DEC has asked the researchers to build a knowledge and question-answering software, similar to those built for biology, in the domain of distributed computing environments.

15.4.6 Future College-Level Biology ITS

The long-term objective is to build tutoring systems for entire courses that compete well with human tutors. In the meantime, researchers will build and evaluate the core components of a tutoring system that competes well with textbooks for an important portion of a course. The chosen domain is that of plant physiological ecology in an introductory biology course. In addition to introductory material, the system will teach advanced material on the topic that has not been covered in the classroom or assigned readings.

Development Status Prototype system under development.

Architecture To date, the aspect of intelligent tutoring on which the researchers have focused is automatic explanation generation. The system architecture for this generator consists of six components: a query interpreter, an explanation planner and a planning space, a linguistic realizer, a knowledge base, and a user model, as shown in Figure 47 on page 218.

Query Interpreter The explanation generation begins when a query is posed. The query may be furnished by the student of an ITS, by the pedagogical planner of an ITS, or by the user of an advisory system. If the query is posed in natural language, the Query Interpreter translates it to a canonical form in an internal representation.

Explanation Planner and Planing Space Upon receiving the query, the Explanation Planner inspects the Knowledge Base to obtain the raw materials for its explanation. A data structure called a Planning Space has been developed to house evolving explanation plans. The system begins constructing an explanation plan by placing the preliminary constituents of the plan into the planning space. These constituents are discourse-elements, which are

sets of propositions to be communicated. Each discourse-element specifies the content of one or more sentences to be generated. The system organized the explanation plan by partitioning the planning space into clusters, each of which will eventually be translated into a paragraph of text.

Linguistic Realizer

Once planning is complete, the planner passes the discourse-elements to the Linguistic Realizer, which translates them into text. Given a discourse-element, the Realizer selects an appropriate syntactic structure and makes all the lower-level linguistic decisions about pronominalization, ellipsis, and lexical choice. Realization is complete when the entire explanation plan (that is, all the discourse-elements of all the clusters) has been translated into a multi-paragraph exposition.

Because text planning precedes realization in this model, it is known as a *pipeline* model. Pipeline models have been adopted by most of the work in multi-sentential natural language generation.

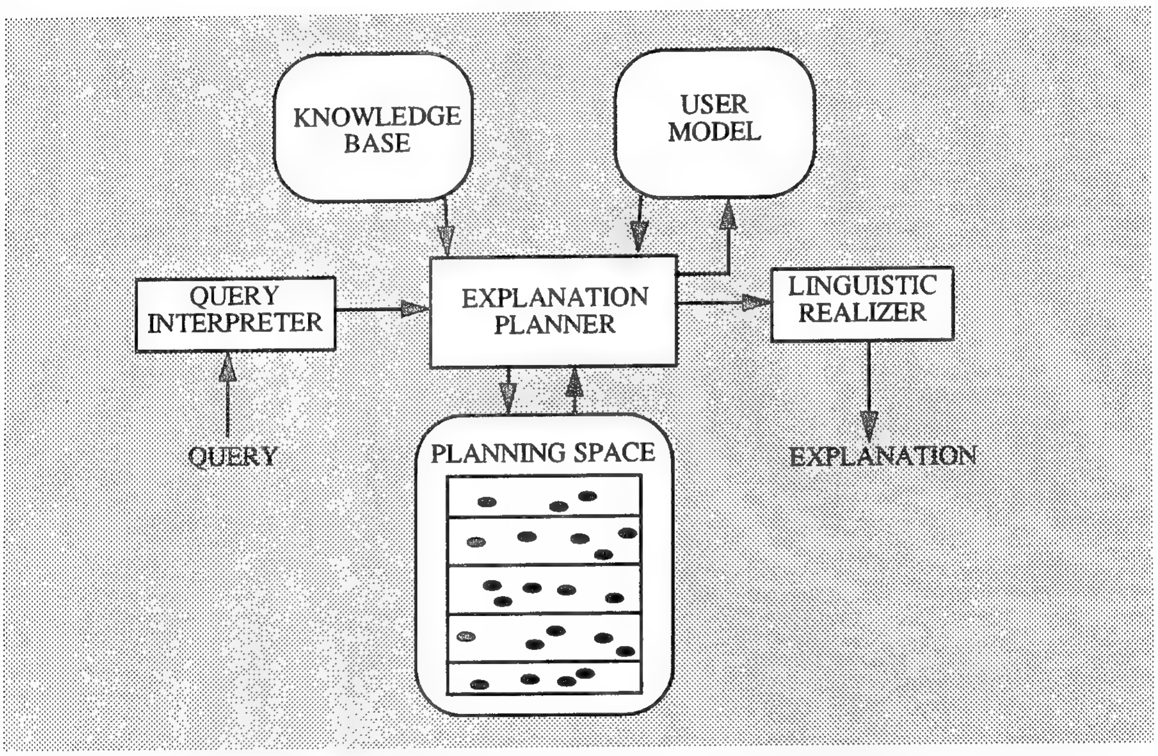


Figure 47. Explanation Generation Architecture

Operating Environment

The existing software is written in Lucid Common Lisp and CLIM, and has been ported to Common Lisp provided by most other vendor (such as Allegro, Symbolics, and Mac-Lisp). The current hardware platform is the DEC 5000 workstation, though any workstation running Unix can be used. The development environment will soon be moved to DEC Alpha workstations.

Evaluation Status

The tutoring system will be evaluated by using it to teach at the University of Texas at Austin. The evaluation will be based on data from the following experiment. Students will be paid to spend extra time in the course studying the advanced mate-

rial with the help of the tutoring system. When the students are comfortable using the system, they will be given several assignments. Each assignment will require answers and explanations for a range of technical questions on both the introductory and advanced material. To complete their assignments, the students will be randomly assigned to three groups. Students in the "traditional" group will be permitted to use any standard (non-human) resources, such as textbooks and laboratory equipment. The "tutoring" group will be allowed to use only the tutor, and the "eclectic" group will be allowed to use both traditional sources and the tutor.

The performance of the three groups of students will be compared on correctness and completeness of answers and on efficiency of task completion. The students' answers and explanations will be judged by the teaching staff for the biology course, who will not be apprised of the students' identity or group. If a benefit for the tutoring system is found, researchers will separately analyze student performance on the introductory material to see if a benefit exists even when the material has been covered in the classroom. Including the eclectic group will further allow the researchers to ascertain whether there is a synergistic effect among the three sources of information—classroom, textbook, and tutoring system. The students' proficiency in terms of the amount of time used to complete the assignment will be measured, controlling for the correctness of the students' responses. For each of the three groups, the researchers will also measure the students' interest in the advanced materials taught. This assessment will be based on questions from standard course evaluations.

In addition, the researchers will videotape a few students from each of the three groups. Immediately after each session, these students will be shown the videotapes and asked to reconstruct their thinking and impressions as they performed the assignment. These students will not be included in the quantitative measures discussed above.

This experiment will establish a baseline of performance for the tutoring system. In follow-up experiments the researchers will determine the individual contributions of the system's parts. For example, by removing the subsystem that graphically illustrates biological concepts and processes, they can measure the contribution of visualization. Similarly, by replacing a subsystem with one which provides the same general functionality, they can compare competing methods. For example, they can compare the initial, relatively simple approach to student modelling with other approaches described in the literature.

15.5 Planned Projects

15.5.1 Educational Support Systems Based on ESSCOTS Applications Project

Initial work in the development and field testing of microworlds demonstrated that through self-directed inquiry using empowering microworlds, students can often grasp ideas previously thought beyond them, that the microworlds dramatically improve motivation and ownership of discovered ideas, and students can also learn generic inquiry skills.[2]

Research has confirmed that inquiry is a powerful way to learn, when supported by appropriate tools and structure. It has also led to a better model of the cognitive process processes of learning through inquiry, which in turn guides the design of more powerful supporting microworlds. The early work, however, also underscored the high cost of developing inquiry-based learning systems. This project will address the cost issue by developing ESSCOTS that wrap an educational software environment around existing commercial software products. The cost savings that will result from this approach will allow more resources to be spent on important evaluation and implementation problems of inquiry-based learning systems. To date, evaluation and implementation issues have received little attention; most research on inquiry-based learning systems stops with "proof of concept" studies.

In particular, this effort will focus on theoretical and evaluative issues underlying ESSCOTS. It will investigate how inquiry-oriented systems developed as ESSCOTS might result in better student learning because they support key cognitive components of learning in general and of learning through inquiry in particular.

Programmatic Background

RAND Corporation is expected to conduct this project under NSF sponsorship. No grant has yet been awarded.

RAND will establish a group of external project consultants with expertise in computer-based learning and inquiry systems to help guide the work. RAND is also forming a consortium on academic and business partners to support the eventual commercialization of the developed products; at present, ESRI has indicated willingness to become a business partner. Santa Monica High School, Santa Monica, California, will provide a field test site.

Planned Products

The primary products will be the curricula in which ESSCOTS are embedded, along with supporting materials and a teacher workshop. Except for the latter, new network technologies will be used to provide immediate and complete access to these research products. In particular, most of the papers, computer code, data, and video products will be accessible on the Internet through a hypertext navigation tool called Xmosiac.

In addition, research findings will be disseminated through papers presented at major educational conferences (for example, AERA, International Conference on the Learning Sciences), at more specialized conferences on technology and learning (for example, AI and Education), and in talks at universities. Major research milestones will be published in various educational journals and books (for example, Cognition and Instruction, Instructional Science, and Interactive Learning Environments). A final report for the project will summarize all major accomplish-

[2] This previous work was conducted under NSF Grants MDR-8751515 and MDR-9055573.

ments. The creation of at least one narrated videotape of students using each ESSCOTS is anticipated, using footage gathered during piloting and fieldings.

Approach

At least four specific ESSCOTS will be developed to help students learn important cognitive skills and come to grips with key social issues. Instead of doing drill-and-practice mathematics, students will learn through substantial projects that tackle real problems using real tools. For example, they may employ simulations to develop models of complex, perhaps chaotic, real-world phenomena, like the flow of toxic waste streams or the spread of forest fires. These subjects are ones that future professionals, and informed citizens in general, need to know. Basic numeric, symbolic, and social literacy will still continue to be important, but only as prerequisites. In the past, schools have been limited to educating students on these basics. They simply do not have the resources to make real-world problems accessible to large student populations. The authentic tools of science and business, when embedded in ESSCOTS, are intended to bridge this gap.

The main goals of the research follow from the fundamental problems of inquiry learning, current success in confronting these problems, and the ESSCOTS approach. Here the four major goals are stated as tasks.

Task 1: *Identifying COTS.* Previously defined criteria for the evaluation of candidate COTS will be applied to identify at least four COTS that can be the basis for ESSCOTS. Key among the criteria will be that the COTS provide a basis for an ESSCOTS that permits students to learn valuable skills. Hence, the selection of COTS will involve the articulation of the important learning outcomes expected of the ESSCOTS.

COTS currently being considered as candidates include the following.

- ARC/INFO for analyzing social trends using visualization tools.
- Ithink and LabVIEW 2 for systems analysis using powerful simulation tools and graphical front-ends.
- Xmosiac for learning how to navigate “cyberspace” using Internet tools.
- IBIS and gIBIS for argument development, analysis, and documentation.
- Gene Construction Kit for conducting gene-splicing experiments.

Task 2: *Developing ESSCOTS.* Once a COTS has been selected, developing an ESSCOTS means providing any generic inquiry tools the COTS does not supply, developing an appropriate interface for students, and providing the smart agents necessary to help guide students’ inquiries and to overcome the fundamental cognitive problems of inquiry learning. These components will compose the bulk of the “ESS” for each ESSCOTS and will be the main educational software created from scratch in the research. The key principles that will be employed include the development of dynamic adapting student interfaces, and the use of open architectures and reusable inquiry tools.

Task 3: *Evaluating ESSCOTS.* The cost effectiveness of each ESSCOTS will be demonstrated through pilot studies in RAND’s Learning and Technology Research Lab. Tools previously developed to uncover the micro- and macro-structure of inquiry

learning will be used, and improved, to address the central evaluation problems confronting inquiry learning. Evaluation will focus on two distinct kinds of cognitive skills: (1) student inquiry skills, and (2) student topic-specific knowledge. Evaluation will serve additional uses by providing data that will (1) improve the design of ESSCOTS, interfaces, and smart agents; (2) develop classroom curricula and projects; and (3) suggest useful teaching strategies and mentoring roles.

Task 4: *Scaling-Up ESSCOTS.* Each ESSCOTS will be scaled up to address fundamental implementation problems. For each system, an embedding curricula will be developed and field tested at local secondary schools. Depending on their success, the curricula will be made available to classes across the country. In addition to the project-based curricula, supporting materials will be developed, a teacher workshop will be conducted, and the implementation will be evaluated.

16. TECHNOLOGY REINVESTMENT PROJECT

16.1 Mission and Role of the TRP

As stated in [ARPA 1993], the mission of the Technology Reinvestment Project (TRP) is "to stimulate the transition to a growing, integrated, national industrial capability which provides the most advanced, affordable, military systems and the most competitive commercial products."

The TRP is a fully collaborative, multi-agency effort supported by ARPA, the Department of Energy, the Department of Transportation, the Department of Commerce's National Institute of Standards and Technology, NSF, and NASA. Via a Memorandum of Understanding, these agencies have formed the Defense Technology Conversion Council (DTCC) to administer the TRP. The DTCC is chaired by ARPA and is responsible for coordinating and integrating Federal Executive Branch activities for technology reinvestment.

The project strategy is to perform the following tasks:

- Develop technologies which enable new products and processes.
- Deploy existing technology into commercial and military products and processes.
- Stimulate the integration of military and commercial research and production activities.

16.2 Overview of TRP

Activities fall into three broad areas: Technology Development, Technology Deployment, and Manufacturing Education and Training. The goal of the Technology Development activity area is to promote the development of dual-use technologies. Activities deal with the creation of new product and process technologies and exploration of their potential for commercial and/or defense applications. The Technology Deployment activity area is concerned with establishing links to existing technology capabilities for small and medium-sized businesses. The goal of the third activity area, Manufacturing Education and

Training, is to improve the general state of US competitiveness and productivity, and provide a high quality work force for the next century.

Technology Development Activity Area:

- Information Infrastructure
- Electronics Design and Manufacturing
- Mechanical Design and Manufacturing
- Materials/Structures Manufacturing
- Health Care Technology
- Training/Instruction Technology
- Environment Technology
- Aeronautical Technologies
- Vehicle Technology
- Shipbuilding Industrial Infrastructure
- Advanced Battery Technology

Manufacturing Education and Training Activity Area:

- Engineering Education in Manufacturing Across the Curriculum
- Practice-Oriented Master's Degree Programs
- Retraining the Manufacturing Work-force
- Educational Traineeships for Defense Industry Engineers
- Manufacturing Engineering Education Coalition
- Supplementary Education Awards to Ongoing Centers and Coalitions Devoted to Manufacturing
- Individual/Group Innovations in Engineering Education in Manufacturing

Figure 48. Focus Areas for the FY93 TRP Competition

The legislation for TRP funding requires that all proposals be selected competitively. The first competition was held in FY93 and proposals were invited for all three activity areas, with specific focus areas as shown in Figure 48 on page 224. In FY94, the TRP plans to conduct two separate competitions. The first is termed the Focused Competition. The scope of this competition is limited to the Technology Development and Technology Deployment activity areas. The specific focus areas for this competition are identified in Figure 49 on page 225. Proposals are due by the end of June 1994, and the announcement of proposals selected for negotiation is expected before the end of FY94. The second FY94

competition is expected to be announced in early Summer 1994. Specific technology topics have not yet been selected for this competition, but will address all three activity areas.

Technology Development Activity Area:

- High Density Data Storage Systems
- Object Technology for Rapid Software Development and Delivery
- Interoperability Testbeds for the National Information Infrastructure
- High Definition Systems Manufacturing
- Low-Cost Electronic Packaging
- Uncooled Infrared Sensors
- Environmental Sensors

Technology Deployment Activity Area:

- Manufacturing Extension Centers

Figure 49. Focus Areas for the FY94 Focused TRP Competition

16.3 Future Tasks

Two hundred and twelve proposals were selected for award negotiation in the FY93 competition. Of these, three address ITS and are discussed further below. In each case, contract negotiations are still in progress and work has not yet begun.

16.3.1 Next Generation Authoring Tools and Instructional Applications Project

The goal of this work is to accelerate the development of new authoring tools and instructional applications. Next generation authoring tools will allow design teams to build instructional applications that are engaging, effective, and viable. Success stories indicate that intelligent instructional applications have effectively reduced by one-third to one-half the time required for learning. In addition, success stories indicate that a new generation of authoring tools that can be used by non-programming professionals can reduce the time to develop instructional titles with embedded intelligence. The authoring tools developed here will reduce the time required to develop an hour of instruction by an order of magnitude compared to existing tools, and will all but eliminate the need for programmers on design teams. In two to three years, these authoring tools will generate cross-platform output using the standards set by the Apple-IBM alliance companies, Taligent and Kaleida.

The scope of this work includes authoring tools for several different types of learning environments. One of these is ITS; others include construction kits, performance sup-

port systems, and edutainment systems. The following discusses the project as a whole; it is not restricted to discussion of the ITS-related parts. Section 16.3.2 on page 229 discusses the prototype ITS authoring suite that is the basis of the ITS authoring product to be developed. Prof. Bev Woolf from the University of Massachusetts will be the Principal Investigator for this part of the work.

Programmatic Background

The work will be accomplished through a consortium of industry and university members, with government/military associates: the East/West Consortium. Specifically, the consortium members are University of Massachusetts, Carnegie Mellon University, University of Colorado, Stanford University, Houghton Mifflin, PWS Publishing Company (and sister corporations Wadsworth and Brooks/Cole), and Apple Computer. The project is expected to be a three-year effort, funded at the level of $2M a year by each of ARPA and the East/West Consortium.

Planned Products

In the first 18 months of the contract, the consortium will produce 2 main types of deliverable: task-specific authoring tools and computer-based, intelligent-multimedia instructional applications. Products to be developed in the second 18 months have yet to be specified.

The choice of which authoring tools to develop will be motivated in part by the proven effective learning architectures that provide good coverage for a range of learner-domain-task situations. The choice of which instructional applications to develop is based on four considerations: existing content, the Department of Labor's Secretary's Commission on Achieving Necessary Skills (SCANS) report, needs of test sites, and domain expertise. The expected products are identified in Figure 50 on page 228.

Approach

The approach derives from two observations: (1) No single learning environment or approach to instruction is best and proven effective for all task-content-learner situations. However, a dozen learning environments provide solid coverage. (2) No single authoring tool is best and proven effective for each learning project and design team. However, a variety of task-specific authoring tools will allow authors to select the right tool for the job, and then be productive in the chosen tool.

Prototype authoring tools have already been developed at member universities and at Apple. They will be further developed and refined. In addition, new authoring tools will be developed. These authoring tools, and the collaboration infrastructure, will allow proven effective learning environments pioneered by university members to be rapidly rebuilt and enhanced with existing content. In collaboration with a number of test sites, instructional applications will be developed and evaluated. CD-ROM distribution techniques will be explored.

The effort is structured as a set of six tasks divided into two phases, each consisting of three tasks. (Phase 2 will be contingent on contract renewal.)

Task 1: *Set-Up, Design, Porting*. Computers, ISDN communication lines, desktop video conferencing and media sharing equipment may be installed at some of the sites. Contractors will be hired to assist in the software development, production, and documentation of the authoring tools and titles. All team members, including rep-

resentatives from key test sites, will receive training in the ToolBuilder technology and collaboration infrastructure technologies. Subsequently, design-build teams for the authoring tools will focus on their specific tasks. While the teams produce rapid iterations on their design, parts of existing university systems may be ported into ToolBuilder. ERS (Engineering Requirements Specifications) for the extant systems will be generated. In addition, existing content that may be used in the production of the titles will be indexed and placed on mediaservers. A decision will be made about programming environments, and existing authoring tools will be used for each instructional application.

Task 2: *Development and Recording Productivity Data.* This task will focus primarily on implementation and indexing content in mediaservers. The authoring tools and titles will be pressed into alpha CD-ROM, and a mini-evaluation will be done. Since the goal is to reduce development time to 30 hours for 1 hour training for many instructional applications, productivity data using the authoring tools will be carefully recorded. Productivity data will record the amount of time required for each design step recorded as a function of activity (for example, training of the experts versus development of the tutor) and as a function of participant role (for example, domain expert versus knowledge engineer). Productivity data will provide a cost metric that can ultimately be used in a larger cost-benefit analysis to assess the effectiveness and efficiency of these systems.

Task 3: *Evaluation.* This task addresses implementation in preparation for a formal evaluation. Prior to evaluation, the authoring tools and titles will be pressed into beta CD-ROM. The authoring tools will be evaluated in terms of their ease of use by non-programmers, reduction in development time over traditional tools, and generality for use in other domains. The titles will be evaluated in terms of the degree to which they prove engaging, effective, and how viable they are.

Task 4: *Revisions and Initial Product Handoffs.* The titles and authoring tools will be revised in light of the complete evaluation results. In addition, with the aid of the improved tools, additional instructional applications will be developed. The revised authoring tools and titles, along with on-line documentation, will be pressed into final CD-ROM. Members will meet to demonstrate their progress to relevant product groups in the industry organizations. Product handoff will take place where warranted.

Task 5: *Elaboration and Further Evaluation.* This period will focus on elaborating the authoring tools and instructional applications to cover more of the space of curriculum, individual differences in learning, and tutoring strategies. In addition, the amount of content in the mediaservers that can be used for several purposes will grow. Further evaluations will be conducted.

Task 6: *Final Productization Plan for Prototypes.* This period will allow working with third-party developers o customize the tools for particular vertical market niches, working with ARPA and other organizations to more fully train users of the new authoring tools, final publications of the results of the evaluations, and refining plans for the commercialization of the prototype authoring tools and instructional applications.

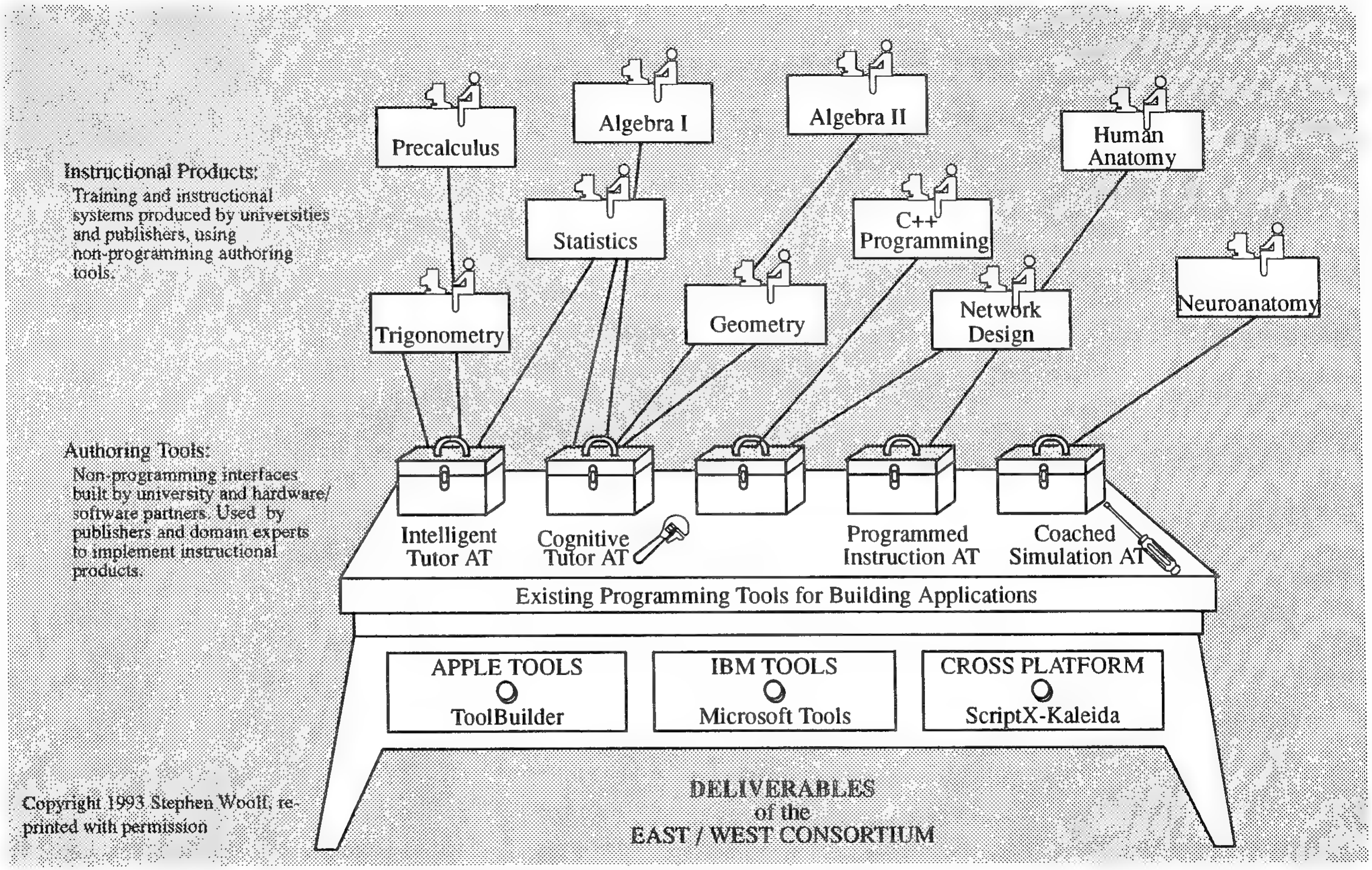


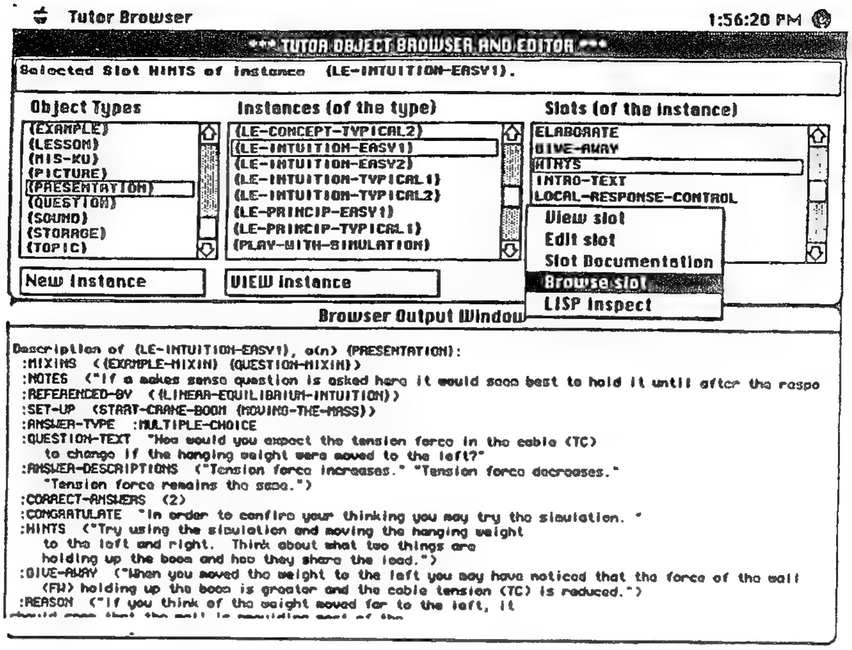
Figure 50. Potential Authoring Tool TRP Products

16.3.2 Knowledge-Based ITS Tools

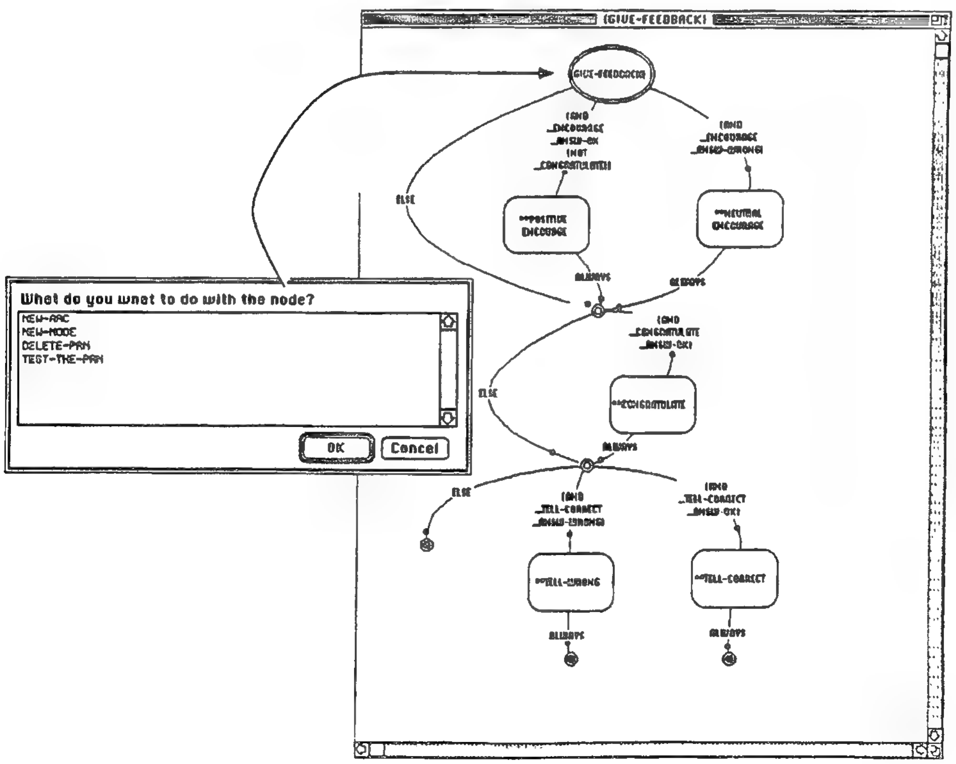
The University of Massachusetts prototype suite of tools enables a high school teacher with only slight familiarity with computers to easily built, test, and modify intelligent tutors [Woolf 1990]. The authoring tools are designed specifically to enable customization of both domain content and teaching style of the tutor. They are domain independent so that tutors for additional subjects can be customized or built using these tools. The prototype tool suite, built with NSF funding, was the first ever used to empirically study (1) the hands-on involvement of educators in building intelligent tutors, and (2) the necessary features of an authoring shell for tutors [Murray 1992]. The formative evaluation of the prototype involved 3 educators over a 16-month period and lab tests with 20 students. Productivity data from this study showed that time required for building a tutor was about 85 development hours per hour of instruction. This figure compares favorably with the estimated 100 to 300 hours required to build traditional computer-assisted instruction. Educators with no experience in computer programming worked with computer scientists to build, evaluate, and modify the computer tutor.

The authoring tools visually portray and support changes in key concepts and in the structure of the domain knowledge. The Domain Browser, shown in the top of Figure 51 on page 230, represents the content and structure of tutoring primitives including, for example, encoded examples, lessons, topics, and presentations. Scrollable listings, pop-up operations menus, and a standard edit window allow a teacher to copy, edit, test, browse, and view components of the domain knowledge base. The Topic Network Editor, shown in the bottom of Figure 51 on page 230, defines and graphically represents textual domain topics and their relationship to each other. Each topic is highlighted during student interaction with the system, and the teacher can inspect and modify nodes and arcs while the system traces the state of the tutor. The Tutoring Strategy Editor in Figure 52 on page 231 represents and traces the flow of control as the tutor reasons about tutoring strategy. Teachers use this editor to create, delete, re-position, and test tutoring strategy nodes during trial runs of the tutor. This editor supports the creation and modification of multiple tutoring strategies. These tools and others not shown constitute a knowledge acquisition shell for representing and rapid prototyping domain and tutoring knowledge.

The commercial suite of tools, including topic network, browser, graphic knowledge structure, and visible student model, will allow non-programmers to rapidly built knowledge-based tutors and enable instructors to monitor the internal state of the tutor while it interacts with the student. Novel methods for visually representing procedural/con-



Domain Browser



Tutoring Strategy Editor

Figure 51. Browser and Strategy Editor

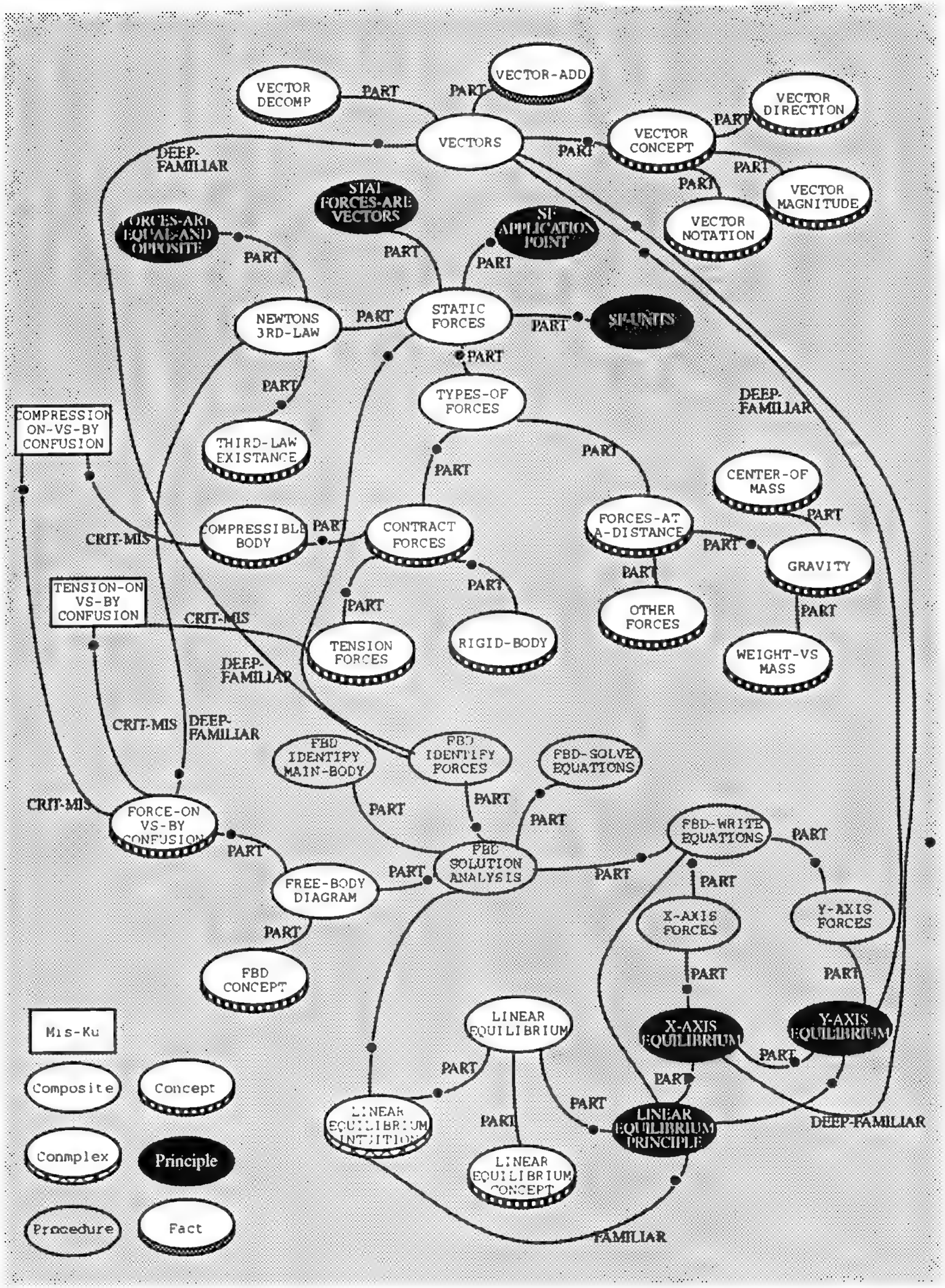


Figure 52. Topic Network

trol knowledge will be implemented. The alpha release tools will undergo formative evaluation prior to refinement and demonstration to potential customers.

The authoring tool suite will be used to develop an ITS for students, with little experience in using mathematics and other problem-solving skills, whose ability level ranges from basic operations (such as addition and subtraction) to beginning algebra. The ITS will incorporate content designed to bolster students' confidence with basic mathematics, providing an environment that is supportive, realistic, and motivating. The tutor will generate a customized learning environment that provides the student with a wide variety of types and levels of problems solving abilities, by tracking the student's performance and responses through a variety of models. It will also provide intelligent individualized evaluation to guide further learning based on the assessment of the student's knowledge, skills, and interest.

16.3.3 Engineering Academy of Southern New England Project

This effort entails designing, implementing, evaluating, and disseminating an integrated system of education for engineers in the next century. Engineers educated to provide leadership on all aspects of the precision manufacturing enterprise are a critical resource. In the heavily industrialized region of southern New England, there is a strong regional industry endorsement of a coordinated approach to manufacturing engineering education in the public engineering schools, and significant industry need for post-BS education of the existing workforce. The articulated agreements among the Schools of Engineering promote institutional interchange of new manufacturing-related instructional approaches, materials, courses, and modules, thereby enhancing the implementation of faculty resources.

Important features of this effort include the following:

- Innovative engineering curricula that integrate process and product development into BS and MS degree programs, and into continuing education opportunities for the regional engineering workforce will be regionally coordinated.
- Comprehensive, vertically integrated educational approaches will involve not only undergraduate engineering programs, but K-12, masters-level programs, and efficient delivery of post-baccalaureate continuing education.
- Accessibility of under-represented populations of minorities and women to these engineering programs will be enhanced through the use of pre-college

bridge programs, existing minority engineering and women in engineering programs and through industry mentoring and internship opportunities.

- Locally developed innovative instructional training systems and software will be development for freshmen and sophomore courses in all Academy engineering schools, using techniques proven to greatly enhance learning efficiency.
- A learning and information network for the delivery, interchange, and dissemination of educational material among institutions and among area industry will be used to involve undergraduates, graduates, community college students, and practicing engineers in curriculum material, case studies of manufacturing-related problems, and special computer-led learning tutorials.
- Practice-oriented courses, delivered via the learning and information network, will be videotaped and distributed to other industries and university classrooms.

Many factors point to the need for increased manufacturing in the curriculum. For example, in addition to training in domain specific knowledge, electrical, and chemical engineers need to understand how to design and specify component parts used in their respective industries. Currently, engineers rarely understand how to design parts for efficiency and cost effectiveness. ITS can support the provision of many of the desired features for an engineering curriculum that supports increased manufacturing education.

The ITS designed for manufacturing engineering will employ on-line dynamic analysis and reasoning about student performance and will generate individualized responses. Two ITS are planned: an Injection Molding Tutor and a Stamping Tutor. They will instruct in Design for Manufacturing, Design for Assembly, Selection of Materials for Design, and Engineering Drafting and Drawing. The instructional systems will include simulation, video, and multimedia. They will encode effective pedagogical techniques, good examples, and explanations along with reasons why each teaching strategy might be used. Students in every division of the engineering department will use the developed tutors and they will be ported to three other universities in the Engineering Academy. The first tutor will be in place in the Freshman engineering classroom in the University of Massachusetts in September, 1994. At this stage, feedback from students and faculty will be used to reimplement and redesign the tutor.

The following discusses the project as a whole; it is not restricted to discussion of the ITS-related part. Prof. Bev Woolf from the University of Massachusetts will be the Principal Investigator for the ITS part of this project.

Programmatic Background

This effort will be performed by the Engineering Academy of Southern New England, a public-private alliance of universities, corporations, and state agencies in the heavily industrialized region of Connecticut. Specifically, those involved include the Universities of Connecticut, Massachusetts, and Rhode Island; the Hartford Graduate Center; Central Connecticut State University; Connecticut Community Technical College System; United Technologies Corporation; Computervision; General Electric Company; Hasbro; IBM/ISSC; Northeast Utilities; Southern New England Telephone; Storch Associates; and Raytheon.

TRP funding for this effort is provided from June 1994 to May 1996. Work actually began January 1994 under university and industry funding. The TRP funding amount is approximately $4M. The universities and industries are providing an additional $8M.

Planned Products

Transferable instructional modules, ITS, and industry-based projects for use throughout the four-year engineering curriculum.

Approach

This information is not yet available.

16.3.4 Retraining the Manufacturing Workforce for the Biotechnology and Biomedical Industries Project

The United States maintains significant leadership in the global economy in biotechnology and the biomedical device field. This leadership will be lost if products for these industries cannot be manufactured competitively. Thus it is critical that a highly trained manufacturing workforce is developed in these crucial areas. The Massachusetts Bioengineering Center (MBeC) will identify 100 individuals with at least a BS in engineering and significant manufacturing experience, who are being laid off from defense-dependent industries, and retrain them with a total immersion in biotechnology-bioengineering. Students will earn a certificate in either bioprocessing or medical devices, and will gain direct industry experience through a practicum. With this education and retraining, these individuals should be well prepared to assume mid- to senior-level positions in the manufacturing sector of biotechnology/biomedical companies to meet a growing need.

A new generation of engineering teaching environments will be developed to facilitate the transition of the defense workers into biotechnology/biomedical engineering. These computer environments will focus on manufacturing practice, synthesis, problem solving, group activities, and integration. High-quality training materials, including ITS, will integrate multimedia digital information and AI techniques, and be tailored for a variety of learners. Advanced computer training systems have proven very effective in the military and industry and will be built to support student activity around complex biotechnology/biomedical manufacturing and industrial situations.

As indicated above, one part of the project requires the development of intelligent tutoring modules for use by the students to assist them in their coursework. The development of these modules is expected to begin in Fall 1994, to complement the course work. As yet, the specific areas of focus for the intelligent tutoring modules have not been determined.

Prof. Bev Woolf from the University of Massachusetts will be the Principal Investigator for the ITS part of this project. The following discusses the project as a whole.

Programmatic Background This effort will be performed by the Massachusetts Bioengineering Center (MBeC), which is made up of the Massachusetts Biotechnology Research Institute, Worcester Polytechnic Institute, and the University of Massachusetts Lowell.

The three-year effort will begin June 1, 1994. The total funding is approximately $1M.

Planned Products Biotechnology-bioengineering curriculum and instructional systems. MBeC expects to develop and offer courses for the first three years with contract funding. Thereafter, the proven value of these courses should encourage biotechnology/biomedical companies to pay tuition remission for their employees enrolled in the program.

Approach The three operational areas of administration, curriculum development and delivery, and assessment/dissemination will be supported by a number of specific tasks to accomplish the project objectives. These tasks are as follows (the order is not indicative of sequence).

Task 1: *Student Recruitment.* Announce and advertise the retraining program, accept and evaluate applications, and conduct orientation for incoming students.

Task 2: *Retooling.* Conduct a retooling program to provide displaced defense workers with the requisite background to enable successful pursuit of the certificate program in bioengineering. This program will principally address the core areas of chemistry, microbiology, and biochemistry.

Task 3: *Develop Courses.* Develop new courses especially tailored to meet the needs of the certificate program. These include bioengineering materials, bioprocess engineering, registration validation and quality control, and chemistry for biotechnology and bioengineering.

Task 4: *Develop Labs.* Expand existing laboratories and develop new laboratories to specifically meet the retraining needs for the certificate program.

Task 5: *Conduct Courses.* Teach the component courses of the certificate program.

Task 6: *Practicum.* Provide practical manufacturing experience in the biomedical and biotechnology industry, supervised by in-company mentors and advised by the certificate program faculty.

Task 7: *Certification.* Award certificates to candidates who meet the certificate program requirements.

Task 8: *Form Advisory Committee*. Recruit and organize the external advisory committee for the retraining program.

Task 9: *Faculty Recruitment*. Recruit the part-time and full-time faculty needed to meet the incremental load presented by the certificate program.

Task 10: *Intelligent Tutor Modules*. Module development by specialized staff, in collaboration with faculty and industry manufacturing experts.

Task 11: *Dissemination*. Reporting of program structure, success, etc., to the academic and industrial communities such that the program may be adopted in part or in full by other geographic areas. Teaching, organizational, and industrial interaction materials will be made available for use by other universities, collaboratives, and industrial groups.

Task 12: *Evaluation*. Design, conduct, and reporting of findings for an evaluation of the program to facilitate modification of the certificate program to best meet the needs of the biotechnology and biomedical manufacturing communities.

Task 13: *Advisory Committee Meetings*. Set agenda, communicate with the external advisory committee, and review MBeC progress.

LIST OF REFERENCES

[AMC 1991] Army Materiel Command. *Artificial Intelligence Technology Master Plan.* August 1991. Unpublished.

[Anderson 1994a] Anderson, J.R., A.T. Corbett, K.R. Koedinger, and R. Pelletier. *Methodology for Decveloping a Practical Tutor: Interim Report on NSF Grant MDR 92-53161.* Carnegie Mellon University, February 1994.

[Anderson 1994b] Anderson, J.R. *Cognitive Tutors: Lessons Learned.* Carnegie Mellon University, May 1994.

[Bovair 1990] Bovair, S., D.E. Kieras, and P.G. Polson. "The Acquisition and Performance of Text-Editing Skill: A Cognitive Complexity Analysis." *Human-Computer Interactions*, 5, (1-48), 1990.

[Card 1993] Card, S.K., T.P. Moran, and A. Newell. *The Psycholgy of Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.

[DOD 1991] DepSecDef Memorandum, "Modeling and Simulation Management Plan," 21 June 1991.

[DOD 1994] DOD Directive 5000.59, January 4, 1994.

[Holm 1991] Holmes, W.M. and A. Piper. *U.S. Army Material Command's Intelligent Tutoring System Technology-Base Plan.* In *Proc. 13th Interservice/Industry Training Systems Conference*, 1991.

[Hume 1993] Hume, G., J.A. Michael, A.A. Rovick, and M. Evens. "The Use of Hints as a Tutorial Tactic." In *Proc. 15th Annual Meeting of the Cognitive Science Society*, Boulder, CO, pp. 563-568, 1993.

[Kieras 1990] Kieras, D.E. "A Guide to GOMS Task Analysis." In *Computer Human Interaction (CHI '91) Tutorial.* 1991.

[Krishnakumar 1991] Krishnakumar, K., D. Sawal, J.E. Bailey, et al. "Synthesis of a Simulator-Based Automated Helicopter Hover Trainer." In *Proc. AIAA Guidance and Control Conference*, New Orleans, LA 1990.

[Li 1992] Li, J., J. Seu, M. Evens, J. Michael, and A. Rovick. "Computer Dialogue System (CDS): A System for Capturing Computer-Mediated Dialogue." *Behavior Research Methods, Instruments, and Computers*, Vol. 23, pp. 535-540, 1992.

[Mostow 1993] Mostow, J., A.G. Hauptmann, L.L. Chase, and S. Roth. "Towards a Reading Coach that Listens: Automated Detection of Oral Reading Errors." In *Proc. 11th National Conference on Artificial Intelligence (AAAI93)*, Washington DC, July 1993.

[Mostow 1994] Mostow, J., S.F. Roth, A.G. Hauptmann, and M. Kane. "A Prototype Reading Coach." To be published In *Proc. 12th national Conference on Artificial Intelligence (AAAI94)* 1994.

[NSD 1992] National Science Foundation, Division of Research, Evaluation and Dissemination. *Applications of Advanced Technologies, Abstracts of Funded Projects Fiscal Years 1990 to 1992*. Draft. March 24, 1992.

[NSF 1993] National Science Foundation. *Guide to Programs in the Division of Research, Evaluation and Dissemination.* September 1993.

[Navy 1993] *TAPSTEM: Training and Personnel Systems Science and Technology Evaluation and Management Report.* Navy Personnel Research and Development Center. October 1993.

[Psot 94] Psotka, J. "Immersive Tutoring Systems: Virtual Reality and Education and Training." Unpublished.

[Regian 1988] Regian, J.W. and V.J. Shute. "Basic Research on the Pedagogy of Automated Instruction." To appear in De Jong, T., H. Spada, and D. Towne (in press). *Use of Computer models for Explication, Analysis, and Experiential Learning*. NATO ASI Series. Springer-Verlag.

[Rovick 1992] Rovick, A.A. and J.A. Michael. "The Prediction Table: A Tool for Assessing Students' KNowledge." *American Journal of Physiology*, Vol. 263 (Advanced in Pysiology Education, 8), pp. S33-S36, 1992.

[Seid 86] Seidel, R.J. and P.D. Weddle (eds.). *Computer-Based Instruction in Military Environments*. New York: Plenum Press, 1986.

[Seid 92] Seidel, R.J. and P.R. Chatelier (eds.). *Advanced Technologies Applied to Training Design*. New York: Plenum Press, 1992.

[Seid 94] Seidel, R.J, and O-C. Park. "Applications of Computer Technology in Instruction and Training: A Research Framework and Examples." Special issue, *Instructional Science: An International Journal of Learning and Cognition*. In press.]

[Shut 94] Shute, V.J., and J. Psotka. "Intelligent Tutoring Systems: Past, Present, Future." To appear in D. Jonassen (ed.), *Handbook of Research on Educational Communications and Technology*. Scholastic Publications, 1994.

[Steinberg 1991] Steinberg, E.R. Computer-Assisted Instruction: *A Synthesis of Theory, Practice, and Technology*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[Swar 92] Swartz, M.L. and M. Yazdani (eds.). *Intelligent Tutoring Systems for Foreign Language Learning*. New York: Springer-Verlag. 1992.

[Zacharias 1992] Zacharias, G.L., E.W. Riley, and P.G. Gonsalves. *Intelligent Flight Trainer for Initial Entry Rotary Wing Training*. Charles River Analytics, Inc. Document R91221. September 1992.

LIST OF ACRONYMS

AASERT	Augmentation Awards for Science and Engineering Research Training
AAT	Applications of Advanced Technology
ACC	Air Combat Command
AETC	Air Education and Training Command
AFOSR	Air Force Office of Scientific Research
AFSOC	Air Force Special Operations CommanAFSPACECOM
AFSPACECOM	Air Force Space Command
AHT	Automated Hover Trainer
AI	Artificial Intelligence
AITS	ARPA Intelligent Tutoring System
ARPA	Advanced Research Projects Agency
ASBREM	Armed Services Biomedical Research, Evaluation and Management Committee
AL	Armstrong Laboratory
AMC	Army Materiel Command
ARCOVA	American Research Corporation of Virginia
ARI	Army Research Institute
ARIARDA	Army Research Institute Aviation R&D Activity
ASTO	Advanced Systems Technology Office
ASVAB	Armed Services Vocational Aptitude Battery
ATHS	Airborne Target Handover System
ATCS	Active Thermal Control System
ATMRU	Advanced Training Methods Research Unit
ATTD	Advanced Technology Transition Demsontration
AWACS	Airborne Warning and Control System
BBN	Bolt, Beranek, & Newman
BFTT	Battle Force Tactical Trainer
BJS	Basic Job Skills
CAT	Cognitive Analysis Tool
C3I	Command, Control, Communications, and Intelligentce

CBD	Commerce Business Daily
CBR	Case-Based Reasoning
CBT	Computer-Based Training
CIC	Combat Information Center
CISCO	Center Information Systems Computer Operations
COFT	Conduct of Fire Training
CRADA	Cooperative Research and Development Agreement
CVCC	Combat Vehicle Command and Control
DOD	Department of Defense
DEC	Digital Equipment Corporation
DIS	Distributed Interactive Simulation
DMD	Digital Message Device
DSR	Digital Systems Resources
DTCC	Defense Technology Conversion Council
DTO	Detailed Test Objective
DVI	Digital Video Interactive
EHR	Education and Human Resources
ELIS	Early Language Intervention System
ERS	Engineering Requirements Specifications
ESSCOTS	Educational Support Systems based on Commercial-of-the-Shelf Software
ETS	Educational Testing Service
EW	Electronic Warfare
FED	Forward Entry Device
FIPSE	Fund for Improvement of Postsecondary Education
FST	Fundamental Skills Training
FY	Fiscal Year
G&CD	Guidance and Control Directorate
GOMS	Goals, Operators, Methods and Selection Rules
GUI	Graphical User Interface
HMI	Human Machine Interface
I-COFT	Individual COFT
I/O	Instructor/Operator
ICALL	Intelligent Computer-Assisted Second Language Learning
ICAT	Intelligent Computer-Aided Training and Tutoring
ICATT	Intelligent Computer-Assisted Training Testbeds

ICBT	Intelligent Computer Based Training
ICOFT	Integrated Conduct of Fire Trainer
IDEA	Individuals with Disabilities Education Act
IEOA	Intelligent Embedded Operator Assistant
IERW	Initial Entry Rotary Wing
IET	Intelligent Embedded Trainer
IFT	Intelligent Flight Trainer
IPS	Instrument Pointing System
ISIS	Instruction in Scientific Inquiry Skills
IVIS	InterVehicular Information System
JDL	Joint Director of Laboratories
JFT	Job Family Tutor
JSC	Johnson Space Center
KOALAS	Knowledgeable Observation Analysis Linked Advisory System
LFA	Low Frequency Active
LFG	Lexical Functional Grammar
LRDC	Learning Research and Development Center
M-COFT	Mobile COFT
MARS	Multistatic Acoustic Receiver System
MI	Military Intelligence
MICOM	Missile Command
MIDI	Musical Instrument Digital Interface
MITE	Multi-Node, Task-Sharing, Expert-Instruction
MITT	Microcimputer Intelligence for Technical Training
MLT	Military Language Tutor
MOS	Military Occupational Specialties
MOU	Memorandum of Understanding
MPP	Main Propulsion Pneumatics
MST	Maintenance Skills Tutor
NASA	National Aeronautical Space Administration
NAWCTSD	Naval Air Warfare Center Training Systems Division
NCTM	National Council of Teachers of Mathematics
NIT	Non-Interactive Written Test
NLP	Natural Language Processing
NORAD	North American Aerospace Defense

NSF	National Science Foundation
NTSC	Naval Training Systems Center
OPL	Operating Procedure Language
PAM	Payload-Assist Module
PARI	Procedure, Action, Result, Interpretation
PD	Payload-Assist Module Deploys
PT	Personal Trainer
PUMP	Pittsburgh Urban Mathematics Project
R&D	Research and Development
R-WISE	Reading-Writing in a Supportive Environment
RAM	Radnom Access Memory
RAPIDS	Rapid Prototype ITS Development System
RDEC	Research Development and Engineering Center
RDC	Research Development Corporation
RDT&E	Research, Development, Testing and Evaluation
RED	Research, Evaluation, and Dissemination
RFP	Request for Proposal
RIDES	Rapid ITS Development System
S&T	Science and Technology
SBIR	Small Business Innovation Research
SCANS	Secretary's Commission on Achieving Necessary Skills
SD	Standard Deviation
SHAI	Stottler Henke Associates, Inc.
SHIFT	SpaceHab Intelligent Familiarization Trainer
SME	Subject Matter Expert
SSC	Space Surveillance Center
STB	Software Technology Branch
STD	Science and Technology Directorate
STRICOM	Simulation, Training and Instrumentation Command
SURTASS	Surveillance Towed Array Sensor System
SVO	Subject, Verb, Object
SWE	Simualted Work Environment
TARGET	Task Analysis Rule Generation Tool
TOC	Tactical Operations Center
TiPS	Tutoring in Problem Solving

TAPSTEM	Training and Personnel Systems Science and Technology Evaluation and Management Committee
TER	Training Effectiveness Ratio
TOD	Tactical Operations Display
TOE	Tutor for Orbital Elements
TOT	Transfer of Training
TRADOC	US Army Training Doctrine Command
TRAIN	Training Research for Autoamted Instruction
TRP	Technology Reinvestment Project
UH-1TRS	UH-1 Training Research Simulator
US	United States
VAPS	Virtual Application Prototyping System
VISTA	Virtual Interactive System for Training Applications
VIVIDS	Virtual Interactive Technologies ITS Development System
VPI	Virginia Polytechnic Institute
VPL	Virtual Physics Laboratory
VR	Virtual Reality
VTT	Verbal Troubleshooting Tests
WOZ	Wizard of Oz
WPS	Word Problem Solving
XVT	Extensible Virtual Toolkit

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
September 1994

3. REPORT TYPE AND DATES COVERED
Final

4. TITLE AND SUBTITLE
Government-Sponsored Research and Development Efforts in the Area of Intelligent Tutoring Systems

5. FUNDING NUMBERS
IDA Central Research Program (CRP) 9000-542

6. AUTHOR(S)
Christine Youngblut

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Institute for Defense Analyses (IDA)
1801 N. Beauregard St.
Alexandria, VA 22311-1772

8. PERFORMING ORGANIZATION REPORT NUMBER
IDA Paper P-3003

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Institute for Defense Analyses
1801 N. Beauregard St.
Alexandria, VA 22311-1772

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, unlimited distribution: January 11, 1995.

12b. DISTRIBUTION CODE
2A

13. ABSTRACT (Maximum 200 words)

The U.S. Government has been sponsoring efforts in the development of intelligent tutoring system (ITSs). Information on current activities was collected from the Department of Defense, Department of Education, National Aeronautics and Space Administration, National Science Foundation, and the Technology Reinvestment Project, to be used as a resource for researchers in the field. Although not uniformly available, information was provided on each project's mission and role, planned products, approach, development status, architecture, operating environment, and future plans.

14. SUBJECT TERMS
Intelligent Tutoring Systems, Computer-Assisted Instruction, Human-Computer Interface (HCI), Graphics User Interface (GUI), Training, Software Tools.

15. NUMBER OF PAGES
266

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR